wyle laboratories

7800 Highway 20 West
Huntsville, Alabama 35806
Phone (205) 837-4411 • Fax (205) 830-2109
www.wylelabs.com

TEST REPORT

REPORT NO.: 45827-01 (2)

WYLE JOB NO.: 45827

CLIENT P.O. NO.: 512395

CONTRACT: N/A

TOTAL PAGES (INCLUDING COVER): 180

DATE: October 5, 2001

QUALIFICATION TESTING

OF THE

iVOTRONIC 2000

DRE PRECINCT COUNTER

For

ES&S
11208 John Galt Blvd.
Omaha, NE 68134

STATE OF ALABAMA
COUNTY OF MADISON }

AL Professional Engineer
Registration No. 16011

Joseph T. Hazeltine, PE, being duly sworn, deposes and says: The information contained in this report is the result of complete and carefully conducted testing and is to the best of his knowledge true and correct in all respects.

Joseph T. Hazeltine

SUBSCRIBED and sworn to before me this 8th day of Oct. 2001

Patricia Phillips SEAL

Notary Public in and for the State of Alabama at Large (pap)

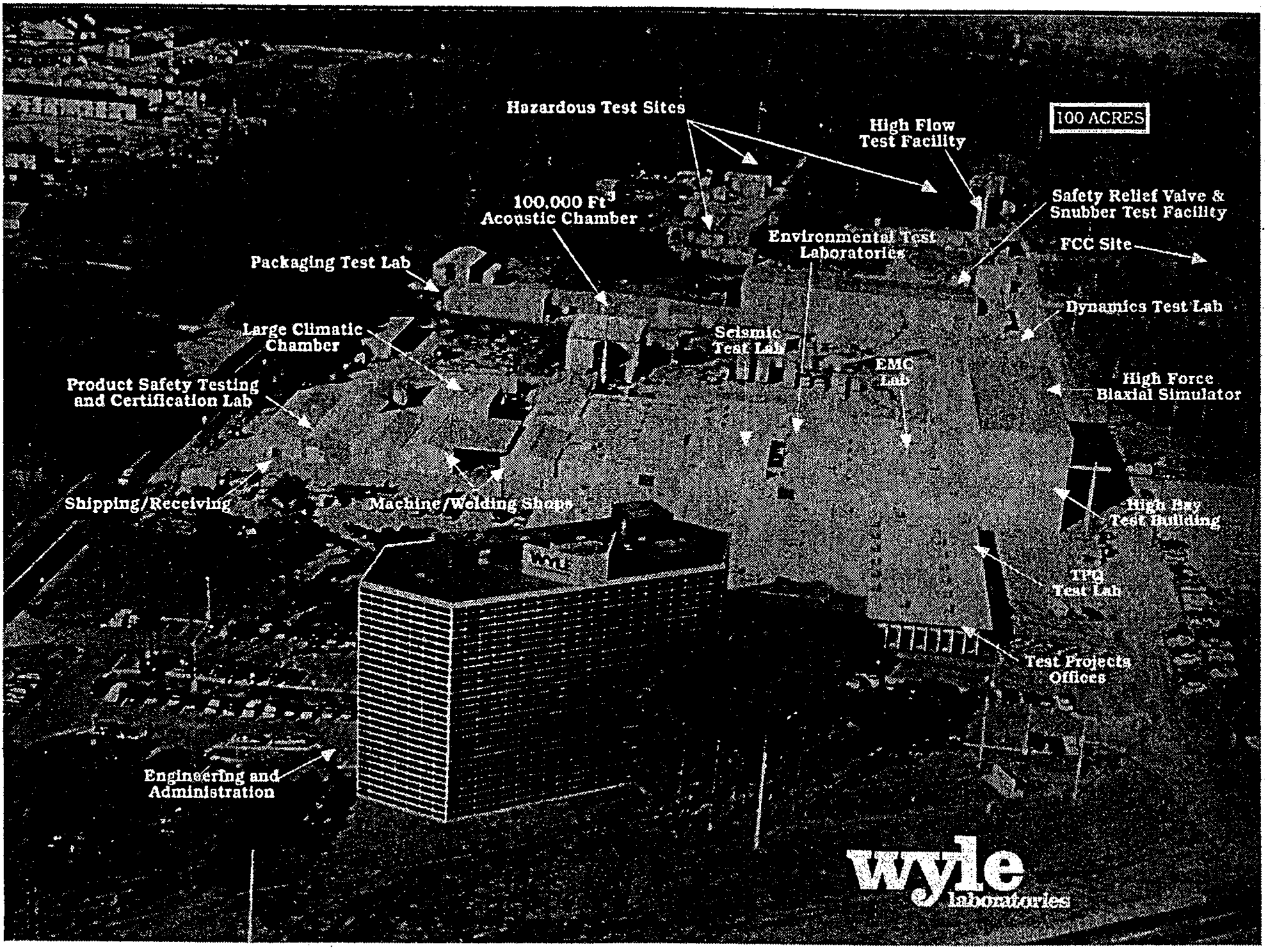
My Commission expires Jan. 16, 2005

Wyle shall have no liability for damages of any kind to person or property, including special or consequential damages, resulting from Wyle's providing the services covered by this report.

TEST BY: 10-8-01
J. R. Dearman, Project Engineer Date

APPROVED BY: Joseph Hazeltine 10/8/01
J. T. Hazeltine, Department Manager Date

WYLE Q.A.: 10/8/01
T. R. Hamilton, Q. A. Manager Date



AERIAL VIEW OF WYLE/HUNTSVILLE

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

ATTACHMENTS

1.0 INTRODUCTION

1.1 Scope

This report presents the test results for Qualification Testing of the ES&S iVotronic 2000 Direct Record Electronic (DRE) Precinct Counter.

1.2 Objective

The objective of this test program was to ensure that the ES&S iVotronic DRE Voting Machine and Voting Machine Firmware, Release 6.1.2, complied with the requirements of ES&S design specifications and with the guidelines of the Federal Election Commission (FEC) National Voting System Standards, January 1990.

1.3 Summary

Qualification testing includes the selective in-depth examination of firmware; the inspection and evaluation of system documentation; tests of hardware under conditions simulating the intended storage, operation, transportation, and maintenance environments; and operational tests verifying system performance and function under normal and abnormal conditions. Qualification testing was limited to precinct-level hardware and resident machine firmware.

The ES&S iVotronic and associated Machine Firmware, Release 6.1.2, was subjected to Reliability and Functional/System-Level Tests. It was demonstrated that the iVotronic and associated Machine Firmware successfully met the qualification test requirements of the Federal Election Commission Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems, January 1990. Qualification testing (in-depth source code review and functional tests) was limited to the firmware and hardware used at the precinct level and did not include any Election Preparation/Central Count subsystems, which typically reside on a personal computer.

Due to the varying requirements of individual jurisdictions, it is recommended by the FEC Standards that local jurisdictions perform pre-election logic and accuracy tests on all systems prior to their use in an election within their jurisdiction.

The following paragraphs address, in greater detail, the design methodology of the ES&S iVotronic of which the ES&S Technical Data Package was the source for much of this information. Additionally, functional and physical characteristics of the iVotronic and the test results observed/recorded during qualification testing are also documented.

2.0 REFERENCES

- ES&S Purchase Order No. 512395
- ES&S Functional Description: ES&S iVotronic 2000 Precinct Counter, Rel. 2.8.
- ES&S iVotronic Hardware Version 1.0, Schematics, 1.0, March 2001.
- ES&S iVotronic Hardware Version 1.0, Assembly Drawings, 1.0, March 2001.
- ES&S iVotronic Hardware Version 1.0, Approved Vendor and Parts List, March 2001.
- ES&S iVotronic Hardware Version 1.0, Bill of Materials, March 2001.
- ES&S iVotronic Hardware Version 1.0, Data Sheets, March 2001.
- ES&S iVotronic Voting System Poll Official's Procedure Manual, Poll Worker Activated Systems, Version 6.1.
- ES&S iVotronic Voting System Operator's Manual, Release 6.1.
- ES&S iVotronic Voting System Operator's Manual, Release 6.1.
- ES&S iVotronic Voting System Poll Official's Procedure Manual, Release 6.1.
- Wyle Laboratories' Quotation No. 545/010002/DB.
- Wyle Laboratories' Job No. 45827.
- Wyle Laboratories' Quality Assurance Program Manual, Revision 2.
- MIL-STD-45662A, "Calibration System Requirements."
- ANSI/NCSL 2540-1, "Calibration Laboratories and Measuring and Test Equipment, General Requirements."
- ISO 10012-1, "Quality Assurance Requirements for Measuring Equipment."

3.0 CUSTOMER

ES&S
11208 John Galt Blvd.
Omaha, NE 68134

4.0 TEST HARDWARE/SOFTWARE DESCRIPTION

4.1 Hardware

4.1.1 iVotronic

The iVotronic Direct Record Electronic (DRE) Voting System consists of two main components: the Personalized Electronic Ballot (PEB) and the Voting Terminal. The exterior shell of both components is of high impact molded plastic. An off-the-shelf thermal serial printer comprises a third external component. Lithium batteries power the memory in both the PEBs and the Terminals. The PEBs are battery-backed RAM storage devices with an infrared window for communication with the terminal. iVotronic Terminals interface with their users via the color LCD display overlaid by a stylus/finger sensitive touch screen. Information is input via the touch screen and output via the screen displays.

The iVotronic terminals also contain an infrared communications window and an electrically isolated three-wire serial port for transmitting data. The serial port is designed to be compatible with a limited subset of system equipment over a short range and under limited conditions, including short cable length. The infrared data link comprises the system communication mechanism between the PEB and Terminal. Magnetic reed switches alert each component of the other's presence when a PEB is inserted. Communications between the system components occur via the proprietary infrared data link, which uses pulse technology and via the proprietary minimal RS232-like serial port.

Program code and election data storage on the iVotronic is accomplished via on-board Flash modules. The base iVotronic system is equipped with 8 Megabytes of FLASH populated in four each of 2-megabyte modules.

Election results are stored in triplicate within the iVotronic terminal. Three identical copies of the election results are redundantly stored on paralleled FLASH modules. During storage operation, the iVotronic will compare the CRC value assigned to each of these modules to ensure no mismatches are present.

The iVotronic is also available in a configuration for use by sight-impaired voters. Through the use of a larger LCD display with audio out, candidate selections can also be made. This configuration's operation was verified during the Qualification testing.

4.0 TEST HARDWARE/SOFTWARE DESCRIPTION (Continued)

4.2 Firmware

4.2.1 iVotronic Firmware

The iVotronic resident firmware is designed to operate with a limited set of proprietary hardware, including the iVotronic Terminal and PEB, and certain off-the-shelf hardware, such as an IBM-compatible PC and a serial printer.

The firmware has been designed to utilize Flash-EEPROMS that stores a complete system audit trail without battery backup. The firmware writes ballot structures in compressed script format, stores votes in complete variable-length images that include write-ins, and records significant events in a fixed-record format.

The iVotronic system processor consists of an x86 family processor.

The iVotronic firmware controls the user controls, indicators and displays.

5.0 PRE-QUALIFICATION TESTS

5.1 Pre-Qualification Test Activities

At the time of submittal for ITA Qualification testing, ES&S had reported the iVotronic had been subjected to extensive in-house testing.

6.0 MATERIALS REQUIRED FOR TESTING

6.1 Software

ES&S provided the latest version of the iVotronic Firmware at the time of ITA Qualification testing commencement. Revisions were made to the firmware during the course of the Qualification Test Program, bringing the version up to Release 6.1.2.upon program completion.

6.2 Equipment

ES&S provided a sufficient number of iVotronic system components to ensure that parallel testing, where feasible, could be performed.

6.3 Test Materials

ES&S provided all ancillary support material required during the course of ITA Qualification testing.

6.4 Deliverable Materials

ES&S provided the latest versions of all hardware and software specifications and poll- worker hardware and software user/maintenance manuals.

7.0 TEST SPECIFICATIONS

Qualification testing and a documentation review were performed to ensure that the precinct-level iVotronic 2000 Voting Machine and associated machine resident firmware were in compliance with the design and functional requirements contained within the following paragraphs.

The iVotronic was functionally tested, as it would be configured for use in an election precinct.

7.1 Environmental Subsystem

7.1. 1 Shelter Requirements

The iVotronic is capable of being operated and stored in any enclosed and habitable facility ordinarily used as a storage area or polling place.

7 .1.2 Space Requirements

The iVotronic can be set up on a jurisdiction-supplied table. The layout of the iVotronic should neither impede the polling place officials from performing their assigned duties, nor disturb the orderly flow of voters through the polling place. Alternately, the iVotronic can be placed within a foldout privacy voting booth equipped with collapsible legs for storage.

7 .1.3 Electrical Supply Requirements

The iVotronic can be powered via standard 115 VAC, 60 Hz facility power. Additionally, the iVotronic can be operated from six D-size NiMh type batteries. The PEB is powered via an internal lithium battery with a projected life of 8 years.

7. 1.4 Environmental Control

Through Qualification testing, it was demonstrated that the iVotronic is capable of storage and operation in temperatures from -15°F to 150°F and 40°F to 100°F, respectively. Such testing is discussed in greater detail in Paragraphs 16.0 & 17.0.

7.2 Ballot Definition Subsystem

7.2.1 Election Programming/Ballot Generation

The Election Preparation Subsystem used for Ballot Definition and Layout activities operates in a central environment separate from the iVotronic precinct hardware. Note that this report addresses Qualification testing limited to the precinct level hardware (iVotronic DRE) and machine resident firmware only.

7.2 Ballot Definition Subsystem (Continued)

7.2.2 Ballot Installation

The iVotronic is designed as a general-purpose voting device. The iVotronic can be matched to any ballot type. The ballot definition and resulting ballot images as displayed by the iVotronic are received as a group of data files downloaded from the Personalized Electronic Ballot (PEB).

7.2.3 Programming and Software Installation

In the event that an updated release of firmware is issued by ES&S, the executable firmware can be uploaded via a compact flash memory card or directly from a PC via a serial link.

7.2.4 Equipment System Readiness Tests

Upon insertion of a PEB, the iVotronic will display the time and date on the screen display as well as the programmed precinct name. The user is then given the opportunity to continue opening the polls. Upon opening, the iVotronic will provide an opportunity for the user to print out a zero's report tape, which includes the results of the system diagnostics.

7.2.5 Verification at the Polling Place

The iVotronic, as a standard required step in the opening of polls, includes a diagnostics pass notification upon successful completion of the power-on-self tests, a zero proof report, the election's identification data, the iVotronic Terminal serial number, and the PEB serial number used to open the system. Additionally, it includes a listing of all offices and measures, and their candidates and responses with the vote counter value of each.

7.2.6 Opening the Polling Place

Opening of the polls on the iVotronic requires the use of a PEB. The PEB is inserted into the iVotronic. An infrared communications device on the front center of the PEB facilitates the transfer of the electronic election data from the PEB to the iVotronic. The type of data stored on the PEB includes ballot layouts, terminal opening and closing times, terminal serial numbers, and vote totals. When the PEB is inserted, the polling official is provided with information on the iVotronic's screen to verify that the correct PEB has been received, and the information is correct for the precinct before the iVotronic is to be used. The polling official is then given the opportunity to generate a zero totals tape. This Information is printed on an ancillary printer via an RS232 connection to the terminal.

7.2 Ballot Definition Subsystem (Continued)

7.2.7 Party Selection Ballot Subsetting

The iVotronic provides for both open and closed primary voting, depending on the ballot definition data that is loaded onto each machine. If voting a closed primary, the poll worker, through the use of on-screen touch-buttons, selects the party that will be activated for the voter. Alternately, the voter can be provided with a programmed PEB, which will call up only the party for which the voter is allowed to vote. If the primary is open, the voter selects the party from the iVotronic's display, and the appropriate ballot is called up. The voter is prevented from voting for a candidate of another party due to the manner by which the iVotronic displays only the selected party ballot, i.e., the candidates for the other parties are not displayed.

The iVotronic supports ballot subsetting due to geographical boundaries or other criteria. The poll worker selects the ballot subset for each voter providing the voter with the appropriate PEB, and the iVotronic only displays the contests and measures the voter is eligible to vote on.

The ballot subset information is part of the ballot definition data that is loaded onto each machine from the PEB.

7.2.8 Enabling the Ballot

Once a voter's registration has been ascertained, the voter can proceed to any available Voter Terminal. A poll worker must activate the PEB for the voter. Once programmed with the correct ballot, the PEB is given to the voter who proceeds to a voting terminal. Once the PEB is inserted into a terminal, the iVotronic will go through several internal checks and show the public count and protective count for that terminal. Once the voter's ballot displays on the terminal screen, the voter proceeds to vote. An alternate method of terminal activation is via a Supervisor Terminal. With the supervisor's terminal, the PEB is inserted into it, and the polling official selects the appropriate ballot, which is then read into the PEB. The PEB is then taken by the poll worker to the voter's terminal, inserted, and the ballot is downloaded. The voter will then proceed to vote. The use of a separate supervisor's terminal for ballot selection versus ballot selection directly at the voter's terminal is up to the discretion of the user jurisdiction.

The ballot is shown to the voter as a series of display pages. The voter moves between pages using the NEXT and BACK buttons. The firmware requires that each page of the ballot be scrolled through by the voter before the ballot can be cast.

7.2 Ballot Definition Subsystem (Continued)

7.2.9 Candidate and Measure Selection

The ballot presented on the iVotronic's screen provides labeling indicating the race and name of every candidate and the titles of every measure, which can be voted. The casting of votes is performed by the voter using a stylus to touch the candidate/measure to be selected. Proper selection is indicated by an 'x' appearing in the candidate's selection box. Additionally, if desired, the voter may deselect a candidate by pressing the button icon a second time and choosing a different candidate. Successful deselecting of a desired candidate is verified by the removal of the 'x' from the candidate's name.

7.2.10 Write-In Voting

Write-in voting is accomplished by selecting the WRITE-IN button on the ballot, which in turn, displays an alphanumeric keyboard on the iVotronic's screen. The voter then has the opportunity to vote for a write-in candidate by typing in the candidate's name. The voter later decides to deselect the write-in candidate, the voter may do so by deselecting the WRITE-IN button and selecting another candidate for that race or by replacing the write-in candidate with the name of another. Multiple write-ins can be provided for any office, so that the voter may enter as many write-in names as the office's 'vote-for' number. Each write-in on the machine-generated results report is tagged to allow the polling workers reviewing the machine reports at the end-of-day to identify if the same name was 'written-in' more than once for the same race. This applies only if the race allows for multiple write-ins.

7.2.11 Casting a Ballot

When the voter has completed voting the ballot, the casting of the ballot is accomplished with the selecting a lighted red VOTE button at the top of the iVotronic's terminal. The vote saving procedure includes saving the voter's ballot image, incrementing counters for each candidate/measure that the voter selected, and validating that the saves and increments were successful. Once a voter has completed voting and pressed the CAST BALLOT button, the iVotronic will power down until another PEB is inserted for activation.

7.2.12 Public Counter

The iVotronic is equipped with a Public Counter. This counter can only be set to zero as part of the System Reset procedure that is done after polls are closed. The Public Counter is incremented once each time a ballot is cast. The Public Counter value is printed on all Zero Proof and Results Reports, and is accessible for viewing by election officials at all times that the iVotronic is powered.

7.2 Ballot Definition Subsystem (Continued)

7.2.13 Protective Counter

The iVotronic is equipped with a Protective Counter that cannot be reset. This counter is set to zero when each system is manufactured. The Protective Counter is incremented once each time a ballot is cast. The Protective Counter value is printed on all Zero Proof and Results Reports, and is accessible for viewing by election officials at all times that the iVotronic is powered.

7.3 DRE Post-Voting Functions

7.3.1 Closing the Polling Place

The iVotronic will not allow the polls to be closed until the official closing time. When a terminal's real time clock recognizes the time as the correct allowable closing time, the terminal displays the message "Do you want to CLOSE this terminal?" To continue, the poll worker inserts a Master PEB into the opened voter terminal. The poll worker is questioned by the terminal if poll closing is to continue. Selecting "yes" results in a message to the poll worker that the PEB is not to be removed until a message allowing such is observed. During this time, votes on the terminal are being transferred to the PEB. This process continues with each PEB used in the same voting precinct.

7.3.2 Obtaining Machine/Polling Place Reports

Using one terminal, and after having connected the printer to the terminal's RS232 port, the poll worker is given the opportunity to print the precinct totals. The terminal will provide the poll worker with the option to print out as many machine totals reports as needed. This report uses the same format as the zero proof report, and shows the following Information:

- the election's identification
- the iVotronic Terminal's serial number
- the PEB SIN
- the firmware release version
- a listing of all offices and measures, and their candidates and responses, with the vote counter value for each

7.0 TEST SPECIFICATIONS (Continued)

7.4 Overall System Requirements

7.4.1 Security

The iVotronic System, through a series of codes and checks, provides for security. The iVotronic requires that all election equipment have the same qualification code. This code secures equipment for each election. Each precinct has a unique code that does not allow ballot and election date to be transferred between polling locations. The precinct code secures the polling location. Because a single PEB is used to open the polls, collect the votes, close the polls, and generate results, it records the serial number of every terminal opened and closed, and the time the opening and closing took place.

7.4.2 Accuracy and Integrity

All communications between iVotronic Terminals and PEBs are CRC checked. PEB's store data in RAM, which is CRC block checked. Terminals store data in three independent memories that are copy checked against each other.

7.5 Hardware Standards

7.5.1 Enclosure

The iVotronic is intended to be used either on a tabletop or free standing. In the free-standing configuration, the iVotronic uses a portable voting booth equipped with collapsible legs for ease in set up, teardown, and storage.

7 5.2 Activity Indicator

There are four individual Voter Notification Screens that provide the voter with the status of their vote, whether it is too many candidates selected for a race, a message stating that all pages of the ballot must be viewed before ballot casting, and also a completion notification that the ballot has been recorded. The iVotronic also has an audible ballot/inactive terminal alarm. This alarm sounds when a voter removes their Voter PEB prior to casting the ballot or when a Voter Terminal remains inactive for 10 minutes with an uncast ballot.

7.5.3 Recording Speed

Voters may make selections and cast ballots as rapidly as they are prepared to do.

7.5.4 Recording Reliability

The iVotronic adequately demonstrated its ability to sustain accuracy during the collection and retention of voting data. It was demonstrated during the Environmental Operating Test that the system was capable of collecting and retaining votes without error over a combined operational period of 163 hours.

7.0 TEST SPECIFICATIONS (Continued)

7.6 DRE Processing Subsystem

7.6.1 Processing Speed

The iVotronic demonstrated adequate response time to operate at speeds sufficient to respond to any operator and voter input without a perceptible delay.

7.6.2 Processing Accuracy

Processing of election data, including audit log data, was performed during Qualification testing to ensure that the iVotronic had the ability to process such data error free. The processing of summary reports and audit log data was limited to those reports as generated by the iVotronic at the precinct level.

7.6.3 Memory Stability

Lithium batteries are used to power the system memory and real time clock and have an expected life expectancy of seven to eight years.

7.7 Reporting Subsystem

7.7.1 Removable Storage Media

The iVotronic uses a Personalized Electronic Ballot (PEB), which is a hand-size cartridge that includes an infrared communications window that stores ballot information and other election data. PEBs activate the iVotronic terminals and loads the ballot onto Voter Terminals for voting.

7.7.2 Printers

The iVotronic uses a printer for printing out zero and totals reports at the precinct level. This printer is capable of generating alphanumeric data to support all reports.

8.0 PHYSICAL CHARACTERISTICS

8.1 Size

The standard iVotronic measures 16.4" by 13.6" by 2.7" and weighs approximately 9 pounds. The iVotronic is classified as Portable equipment (i.e., equipment typically installed and operated on a table or stand to which it is not permanently affixed). The PEB measures 3.6" by 3.8" by 1.1" and weighs 6 ounces.

8.0 PHYSICAL CHARACTERISTICS (Continued)

8.2 Transport and Storage

The iVotronic is self-contained, including a protective padded nylon case for handling and transport. It was demonstrated during Qualification testing that this delivery arrangement provides adequate protection to the iVotronic in the event the machine is dropped during handling from heights of 12 inches. It was demonstrated that when the iVotronic is adequately packaged within a fiberboard shipping container, it could sustain drops from a height of 48 inches.

8.3 Physical Security

The iVotronic is affixed with seals, which must be broken if access to the screws that hold the two halves together are accessed. Additionally, the carrying case can be secured with numbered seals, which must be cut for removal of the iVotronic from its case.

8.4 Transportability

The iVotronic is capable of being transported by road, rail, or air.

9.0 DESIGN, CONSTRUCTION, AND MAINTENANCE CHARACTERISTICS

9.1 Materials, Processes and Parts

The ES&S technical data package contained a listing of those system elements that make up the iVotronic, as well as an assembly procedure.

9.2 Durability

The commercial construction standards observed to be associated with the iVotronic suggest a continued life of at least eight years through normal election use.

9.3 Reliability

A Mean-Time Between Failure of a minimum of 163 hours was demonstrated during Qualification testing as accumulated on four iVotronic machines. This testing was performed during varying temperature and input voltage conditions, and is discussed in further detail within paragraph 16.0. There were no hardware failures observed that resulted in the loss or unacceptable degradation of one or more machine functions during this test.

9.0 DESIGN, CONSTRUCTION, AND MAINTENANCE CHARACTERISTICS (Cont'd)

9.4 Maintainability

The iVotronic is designed with good layout practices for access to internal components. The iVotronic provides a diagnostic capability as a standard part of its operating firmware. These diagnostics include power-up self-tests, and continuous tests in the background while operating. It is the intent of ES&S that an Election Technician would be responsible for handling tasks such as touchscreen calibration, battery changeout, testing and replacement, and module level repair. A module-level repair would be, for example, swapping out a power supply or LCD/touchscreen assembly.

9.5 Electromagnetic Radiation

The iVotronic Voting Machine was subjected to electromagnetic emissions measurements to ensure that it meets the limits for a FCC Part 15, Class B, computing device. The results of the FCC Part 15 testing are documented in Wyle Test Reports 46229 01 and –02, contained in Attachment M.

9.6 Workmanship

ES&S states that workmanship on the iVotronic has been designed with good workmanship practices. Inspection of the iVotronic's interior suggests that good workmanship practices were in use during its manufacture.

9.7 Interchangeability

The iVotronic is designed as a group of subsystems, with well-defined mechanical and electrical interfaces, maximizing interchangeability.

9.8 Safety

The iVotronic was subjected to a product safety review to ensure compliance with UL 1950, Safety of Information Technology Equipment, including Electrical Business Equipment, and was found to comply with no modifications required. The results of the product safety review are documented in Wyle Test report 45827-02 contained in Attachment L.

9.9 Human Engineering

9.9.1 Controls and Displays

All user operations follow a single path, from loading a ballot, through pre-election hardware checkout, accuracy testing, Election Day voting, and post-election functionals.

User prompts are presented in an unambiguous manner.

9.9 Human Engineering (Continued)

9.9.1 Controls and Displays (Continued)

All iVotronic displays and controls are easily accessible.

The voter interacts with the iVotronic only through the LCD touchscreen. Instructions, when needed, are provided on-screen.

Depending on the function, both the poll worker and voter interact with the iVotronic through the main touchscreen.

All messages for the poll worker are shown on the iVotronic's touchscreen.

All displays are readable by persons with normal eyesight. An alternate configuration (15-inch version) of the iVotronic with a larger LCD display screen and audio output capabilities is available to assist sight-impaired voters.

Significant machine operations, such as activating for a voter, casting a ballot, and error conditions, are announced by audible tones.

9.9.2 Data Quality Assessment

The iVotronic Software Technical Data Package addresses in depth the methodology by which the resident firmware has been built to perform real-time monitoring of system status and data quality.

10.0 SOFTWARE STANDARDS

10.1 Vote Recording Accuracy and Integrity

During testing, the iVotronic was subjected to several test elections. During test, all votes were accurately recorded from each ballot cast and accurate summary reports were generated.

Attachment A contains a typical iVotronic machine-generated printout.

10.2 Data and Document Retention

The iVotronic stores all vote data in three separate memory chips. Each of the chips contains a complete record of all ballots that were cast on that Voter Terminal.

10.0 SOFTWARE STANDARDS (Continued)

10.3 Ballot Interpretation Logic

Through simulated elections performed during Qualification testing, the iVotronic demonstrated proper ballot interpretation of the following:

1. Closed and open primary elections
2. Partisan and non-partisan offices
3. Straight party voting
4. Split precincts
5. Vote for N of M
6. Undervotes
7. Total blank ballots
8. Write-In Voting

11.0 SYSTEM AUDIT

11.1 Operational Requirements

The iVotronic provides an audit log for recording all significant system activities. Type of entries include:

Election day activities
Post-election activities
Report printing
Operator induced errors
System errors

11.2 Time, Sequence, and Preservation of Audit Records

The iVotronic incorporates a real-time clock. All audit log entries are automatically stamped with the current date and time. It is not possible to interrupt or disable the logging of events to the Event Log. The audit log contents are accessed via the central election management system software. The audit log cannot be printed directly at the iVotronic level.

11.3 Error Messages

iVotronic errors are reported as they occur and an Event Log entry is made as a standard part of the error response.

When in an error condition, the iVotronic is designed to limit allowable actions to only those that will recover from the error. No extraneous inputs are accepted.

11.0 SYSTEM AUDIT (Continued)

11.4 Status Messages

The iVotronic reports status information to the user immediately in easily understood text. Significant changes in status, such as opening and closing polls, are logged to the Totals tape.

11.5 In-Process Audit Records

The iVotronic is designed to capture all the events and activities normally encountered during in-process activities. Additionally, the in-process record includes each time a vote is cast. A complete listing of audit log entries is contained within the appendices of the user's manual, including error codes where applicable.

11.6 Vote Tally Data

The Vote Tally Data on the iVotronic includes the following separately accumulated items:

- number of ballots cast for each ballot style
- number of ballots cast within each precinct and/or party; candidate and measure totals for each contest
- write-in names that were entered, with tagging to which they were cast; and an associated ballot image record

The latter data is presented to ensure that in multiple vote-for contests, voters do not attempt to cast multiple write-in votes for the same candidate. The write-in name data is stored in randomized order. The Ballot Image contains an exact record of each selection made for each voter stored in a randomized order. This detail provides the ability to account for all undervotes. Overvotes are not possible on the iVotronic.

12.0 SECURITY

12.1 Access Control Measures

Access is controlled by password operation.

12.2 Physical Security Measures

12.2.1 <u>Hardware</u>

Secure storage of the iVotronic will be a function of the user jurisdiction.

12.2 Physical Security Measures (Continued)

12.2.2 Software

The iVotronic contains only firmware. Each jurisdiction must safeguard the physical security of the iVotronic system. The ballot definition data that is created for each election contains no executable code but consists of data structures, templates, and text. The iVotronic does not contain any compiler or similar code. The iVotronic does not contain any self-modifying code.

13.0 QUALITY ASSURANCE

13.1 Quality Control

ES&S provided a copy of the iVotronic Quality Assurance Procedures. The procedures describe the iVotronic assembly and checkout processes as performed by the manufacturer prior to delivery.

13.2 User Documentation

During Qualification testing, ES&S provided copies of the iVotronic Operations. Pollworker and Maintenance Manuals.

14.0 SOFTWARE SYSTEM FUNCTIONAL TESTS

14.1 Software System Functional Test Procedures

The iVotronic was subjected to a series of functional Software System Tests to verify proper operation of the machine as dependent upon the proper performance of the machine's operating software. This included the tasks identified below.

- Test operations performed prior to, during, and after processing of ballots, which included:
  (a) logic tests to verify interpretation of ballot styles to be processed
  (b) accuracy tests to verify ballot reading accuracy
  (c) status tests to verify equipment
  (d) report generation to produce test output data
  (e) report generation of product audit data records
- Procedures applicable to equipment used in the polling place for:
  (a) opening the polling place and enabling the acceptance of ballots
  (b) maintaining a count of processed ballots
  (c) monitoring equipment status
  (d) verifying equipment response to operator input commands
  (e) generating real-time audit messages
  (f) closing the polling place and disabling the acceptance of ballots
  (g) generating election data reports.

14.0 SOFTWARE SYSTEM FUNCTIONAL TESTS (Continued)

14.2 Software System Functional Test Results

The iVotronic Operating Firmware, Release 6.2.1, was successfully subjected to the above System Software Functional Tests.

The iVotronic was subjected to several different test elections. The polls were opened and the machines were enabled for voting. Various votes were cast for each of the ballot styles defined. Following the completion of each test election, the polls were closed and election results (summary reports) and audit log trails were generated. The election results were compared against the predetermined votes cast to ensure that proper ballot logic and accuracy in recording the votes had been obtained.

The iVotronic was additionally subjected to a high-volume-vote reliability/accuracy test.

15.0 SYSTEM LEVEL TESTS

15.1 Volume

During volume testing, four iVotronic systems were subjected to high volume ballot processing and vote recording. Specifically, the iVotronic systems were configured using an automated script to cast and record in excess of 297,589 votes without error. Additionally, in accordance with the FEC guidelines, the iVotronic systems were subjected to reading in excess of the minimum required 5,000 ballots for a precinct counter. No anomalies were encountered. All vote totals were accurate and complete. To ensure the accurate retention of voting data in the unlikely event that a memory storage device fills to its maximum limit, tests were performed to verify the iVotronic response. In each case, the iVotronic disallowed the casting of a ballot when presented with a situation where the storage medium was at capacity and would not be able to successfully capture and store the ballot data.

15.2 Security Tests

As previously described, the iVotronic election definition is secured within the machine by tamper-proof seals or other sealing devices as judged appropriate by the using jurisdiction. Access to the supervisor functions is password protected.

15.3 Usability

Setup and subsequent operation of the iVotronic was found to be relatively straightforward following training. Poll worker operation is straightforward, including the opening and closing of polls, all of which are performed through touchscreen entry/button actuation.

15.0 SYSTEM LEVEL TESTS (Continued)

15.4 Recovery

The iVotronic was successfully tested to verify its ability to recover from certain error-handling conditions. These included the purposeful insertion of incorrect PEBS, power loss, etc.

15.5 Performance

As detailed in other sections of this report, the iVotronic was successfully subjected to several simulated elections to verify poll opening, voting, and poll closing sequences as well as voter recording accuracy and correct ballot logic interpretation

16.0 OPERATING TEST

16.1 Operating, Environmental Test

To demonstrate a minimum acceptable Mean-Time-Between-Failure threshold of the iVotronic, four machines were placed inside an environmental walk-in test chamber and connected to a variable voltage power source. The temperature inside the chamber and the voltage supplied to each of the iVotronics were varied from 40°F to 100°F and from 105 VAC to 129 VAC. The environmental test profile and Chamber Thermal Circular Charts are presented in Attachment C.

Four machines were used during the Operating Test to accumulate the time required to demonstrate a minimum Mean-Time-Between-Failure (MTBF) of 163 hours. By subjecting four machines to the Operating Test environment, the accumulated test time per machine was 40.5 hours. However, testing was continued until the 48-hour chamber cycling had completed. There were no machine hardware anomalies observed during the hardware reliability testing.

17.0 NON-OPERATING ENVIRONMENTAL TESTS

The iVotronic was subjected to various Non-Operating Environmental Tests. Prior to and immediately following each test environment, the iVotronic was powered and subjected to operability functionals to verify continued proper operation. The iVotronic was not powered during the performance of any of the non-operating tests.

17.1 Transit Drop Test

The iVotronic was subjected to a Transit Drop Test. It should be noted that the National Association of State Election Directors (NASED) has substituted Military Standard 810 for the FEC Voting Systems Standards' Drop Test included in the original Standards as published in 1990. The iVotronic was subjected to a baseline operability checkout to verify system readiness. The iVotronic was packaged in a

17.1 Transit Drop Test (Continued)

fiberboard container. The iVotronic and container, weighing less than 100 pounds with dimensions less than 36 inches, resulted in a drop height of 48 inches. The packaged iVotronic was dropped on each face, edge, and corner for a total of 26 drops. Following the drop testing, the iVotronic was removed from its shipping container, powered, and continued operation verified.

Additionally, the iVotronic, while packaged in a padded nylon carrying case, was subjected to eight corner drops, each from a 12-inch drop height. Upon test completion, the iVotronic was powered and continued operation verified.

Attachment D contains a Drop Test Data Sheet.

17.2 Low Temperature Test

The iVotronic was subjected to a Low Temperature Test.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was placed in an environmental test chamber. The chamber temperature was lowered to -15°F and allowed to stabilize. Upon temperature stabilization, the temperature was maintained for an additional four hours. The temperature was then returned to standard laboratory ambient conditions at a rate not exceeding 10°F per minute. The iVotronic was removed from the chamber and inspected for any obvious signs of degradation and/or damage. None were observed. The iVotronic was subjected to a post-test operability checkout and continued operability verified.

Attachment E contains a Low Temperature Thermal Circular Chart.

17.3 High Temperature Test

The iVotronic was subjected to a High Temperature Test.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was placed in an environmental test chamber. The chamber temperature was raised to 150°F and allowed to stabilize. Upon stabilization, the temperature was maintained for an additional four hours. The temperature was then returned to standard laboratory ambient conditions at a rate not exceeding 10°F per minute. The iVotronic was removed from the chamber and inspected for any obvious signs of degradation and/or damage. None were observed.

The iVotronic was subjected to a post-test operability checkout and the results were in agreement with those established during the baseline checkout.

Attachment E contains a High Temperature Thermal Circular Chart.

17.0 NON-OPERATING ENVIRONMENTAL TESTS (Continued)

17.4 Vibration Test

The iVotronic was subjected to Vibration Testing.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was strapped to an electrodynamics shaker while stowed in its black carrying case. One control accelerometer was affixed to the shaker table. Vibration and control was performed with an HP5427 Shock/Vibration Controller. The iVotronic was subjected to the Basic Transportation, Common Carrier profile as depicted in MIL-STD-810D, Method 514.3, Category I. The iVotronic was subjected to vibration for 30 minutes in each orthogonal axis. Upon test completion, the iVotronic was removed from its carrying case and inspected for any obvious signs of degradation and/or damage. None were observed. The iVotronic was subjected to a post-test operability checkout and the results were in agreement with those established during the baseline checkout.

Attachment F contains a Vibration Test Data Sheet and Data Plots.

17.5 Bench Handling Test

The iVotronic was subjected to Bench Handling Tests.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was configured as for normal operation or servicing. Using one edge (base of machine) as a pivot, the opposite edge was raised to a height of four inches above the surface and allowed to drop freely. This was performed an additional five times for a total of six drops. The same was repeated for the remaining three base edges for a total of 24 drops. Upon test completion, the iVotronic was inspected for any obvious signs of degradation and/or damage. None were observed. The iVotronic was successfully subjected to a post-test operability checkout.

Attachment G contains a Bench Handling Test Data Sheet.

17.6 Humidity Test

The iVotronic was subjected to a 240-Hour Humidity Test.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic, stowed in its black carrying case, was placed within a Thermotron Humidity Chamber. The iVotronic was subjected to a 10-day humidity cycle in accordance with the procedures as found in MIL-STD-810D, Method 507.2, Procedure I-Natural Hot Humid. Upon test completion, the iVotronic was removed from its carrying case and inspected for any obvious signs of degradation and/or damage. None were observed. The iVotronic was successfully subjected to a post-test operability checkout.

Attachment H contains Humidity Circular Charts.

17.0 NON-OPERATING ENVIRONMENTAL TESTS (Continued)

17.7 Rain Test

The iVotronic was subjected to a Rain/Drip Test.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was placed in a voting booth/transit case. The iVotronic was then subjected to a Rain Test per MIL-STD-810D, Method 506.2, Procedure II-Drip. The unit, non-operating and in a transportable configuration, was subjected to a waterfall rate of 7 gallons/sq.ft/hr. The water was delivered from a height of approximately 3 feet for a period of 15 minutes. At the conclusion of the 15-minute exposure, the equipment was removed and the inspected for any evidence of water intrusion. None was observed. The unit was powered and successfully subjected to a post-test operability checkout.

Attachment J contains a Rain Test Data Sheet.

17.8 Dust Exposure Test

The iVotronic was subjected to a Dust Exposure Test.

The iVotronic was subjected to a baseline operability checkout to verify system readiness. Upon completion, the iVotronic was placed in a voting booth/transit case. The unit, in its stowed configuration, was subjected to a dust exposure per MIL-STD-810D, Section 11-1.1.1. Following test, the iVotronic was removed from the dust chamber and the accumulated dust brushed off. The iVotronic was inspected for any evidence of dust intrusion and damage. None was observed. The unit was powered and successfully subjected to a post-test operability checkout.

Attachment I contains Dust Test Circular Charts.

18.0 TEST EQUIPMENT AND INSTRUMENTATION

All instrumentation, measuring, and test equipment used in the performance of this test program were calibrated in accordance with Wyle Laboratories' Quality Assurance Program, which complies with the requirements of ANSI/NCSL 2540-11 ISO 10012-11 and Military Specification MIL-STD-45662A. Standards used in performing all calibrations are traceable to the National Institute of Standards and Technology (NIST) by report number and date. When no national standards exist, the standards are traceable to international standards or the basis for calibration is otherwise documented.

Attachment K contains Instrumentation Equipment Sheets.

19.0 WYLE QUALITY ASSURANCE

All work performed on this program was completed in accordance with Wyle Laboratories' Quality Assurance Program.

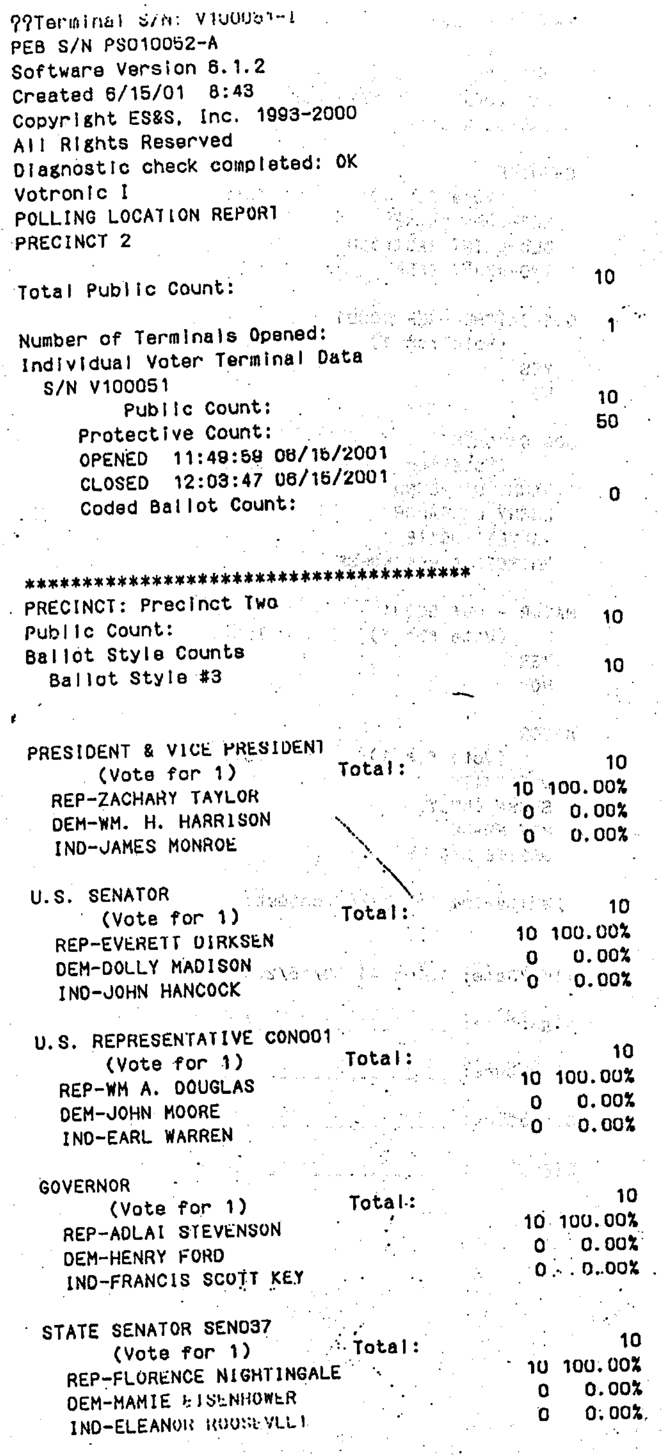
The Wyle Laboratories, Huntsville Facility, Quality Management System is registered in compliance with the ISO-9001 International Quality Standard. Registration has been completed by Quality Management Institute (QMI), a Division of Canadian Standards Association (CSA).

This page intentionally left blank.

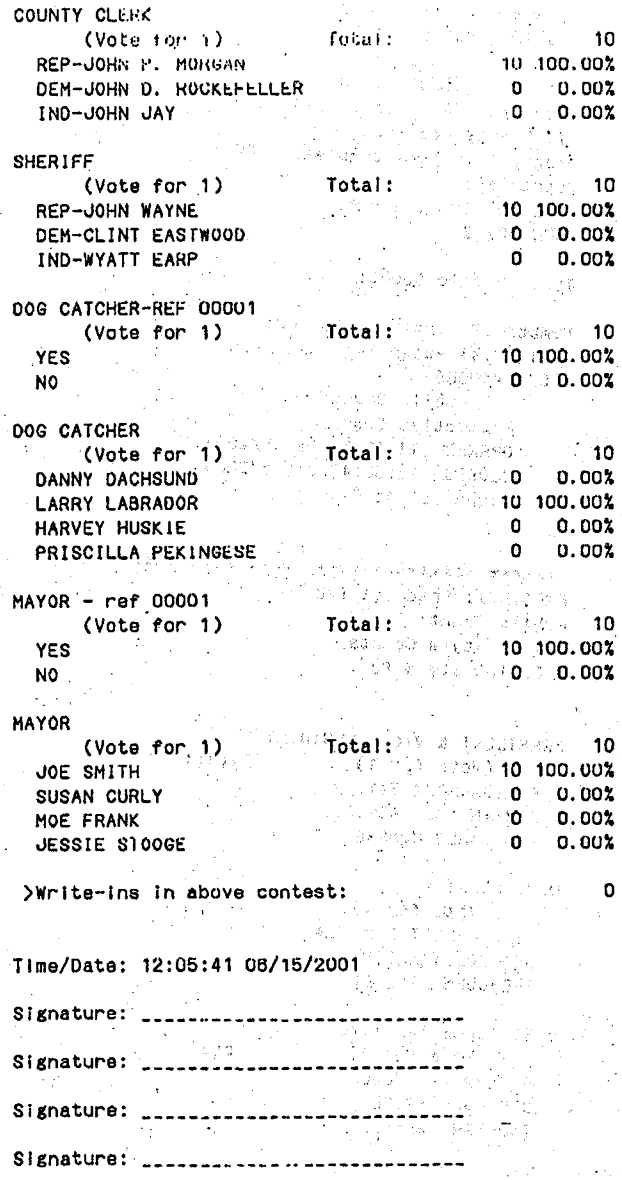
ATTACHMENT A

iVOTRONIC TYPICAL PRECINCT-LEVEL PRINTOUT

This page intentionally left blank.



```
??Terminal S/N: V100051-1
PEB S/N PS010052-A
Software Version 6.1.2
Created 6/15/01  8:43
Copyright ES&S, Inc. 1993-2000
All Rights Reserved
Diagnostic check completed: OK
Votronic I
POLLING LOCATION REPORT
PRECINCT 2

Total Public Count:                              10

Number of Terminals Opened:                       1
Individual Voter Terminal Data
  S/N V100051
          Public Count:                          10
      Protective Count:                          50
      OPENED  11:49:59 06/15/2001
      CLOSED  12:03:47 06/15/2001
      Coded Ballot Count:                         0


**************************************
PRECINCT: Precinct Two                           10
Public Count:
Ballot Style Counts                              10
  Ballot Style #3


PRESIDENT & VICE PRESIDENT
      (Vote for 1)          Total:               10
  REP-ZACHARY TAYLOR                     10 100.00%
  DEM-WM. H. HARRISON                     0   0.00%
  IND-JAMES MONROE                        0   0.00%

U.S. SENATOR
      (Vote for 1)          Total:               10
  REP-EVERETT DIRKSEN                    10 100.00%
  DEM-DOLLY MADISON                       0   0.00%
  IND-JOHN HANCOCK                        0   0.00%

U.S. REPRESENTATIVE CON001
      (Vote for 1)          Total:               10
  REP-WM A. DOUGLAS                      10 100.00%
  DEM-JOHN MOORE                          0   0.00%
  IND-EARL WARREN                         0   0.00%

GOVERNOR
      (Vote for 1)          Total:               10
  REP-ADLAI STEVENSON                    10 100.00%
  DEM-HENRY FORD                          0   0.00%
  IND-FRANCIS SCOTT KEY                   0   0.00%

STATE SENATOR SEN037
      (Vote for 1)          Total:               10
  REP-FLORENCE NIGHTINGALE               10 100.00%
  DEM-MAMIE EISENHOWER                    0   0.00%
  IND-ELEANOR ROOSEVELT                   0   0.00%
```



```
COUNTY CLERK
      (Vote for 1)        Total:              10
  REP-JOHN P. MORGAN                   10 100.00%
  DEM-JOHN D. ROCKEFELLER               0   0.00%
  IND-JOHN JAY                          0   0.00%

SHERIFF
      (Vote for 1)        Total:              10
  REP-JOHN WAYNE                       10 100.00%
  DEM-CLINT EASTWOOD                    0   0.00%
  IND-WYATT EARP                        0   0.00%

DOG CATCHER-REF 00001
      (Vote for 1)        Total:              10
  YES                                  10 100.00%
  NO                                    0   0.00%

DOG CATCHER
      (Vote for 1)        Total:              10
  DANNY DACHSUND                        0   0.00%
  LARRY LABRADOR                       10 100.00%
  HARVEY HUSKIE                         0   0.00%
  PRISCILLA PEKINGESE                   0   0.00%

MAYOR - ref 00001
      (Vote for 1)        Total:              10
  YES                                  10 100.00%
  NO                                    0   0.00%

MAYOR
      (Vote for 1)        Total:              10
  JOE SMITH                            10 100.00%
  SUSAN CURLY                           0   0.00%
  MOE FRANK                             0   0.00%
  JESSIE STOOGE                         0   0.00%

 >Write-ins in above contest:                  0

Time/Date: 12:05:41 08/15/2001

Signature: ____________________________

Signature: ____________________________

Signature: ____________________________

Signature: ____________________________
```

ATTACHMENT B

SOURCE CODE REVIEW SUMMARY REPORTS

This page intentionally left blank.

ESS Software Review Summary

This review covers Version 6.1.2 of the ESS Votronic code with respect to changes made since Version 6.1.1 as well as evaluation of its compliance with the FEC guidelines for software quality and reliability. This evaluation included, but was not limited to, the following considerations:

Readability	How straightforward and apparent was the design?
Understandability	How complicated was the code to implement it?
Modularity	How well was the code divided into logical, functional units?
Robustness	How well does the code handle error conditions or unexpected inputs?
Security	Does the code protect the integrity of voting data at all times?
Maintainability	How easy would it be to extend, fix, or modify this code in the future?
Consistency	Was the design of the code coherent throughout?
Documentation	Does the code contain useful and frequent comments?
Usability	Does the code inform the user about progress or errors?
Flow control	Are control constructs and entry/exit points logical and controlled?

The review report detailed specific instances where it was felt that the code fell short in some area being reviewed, and gave file names and line numbers where applicable to guide the maintainers in making needed corrections. Also included is a general synopsis section where recurring problems were explained and solutions suggested. Also identified are some areas where improvement is needed in non-critical areas, and those changes should be addressed in future versions of this software.

A recommendation is given at the end of this document.

FLASH1.H
FONT.C
FONT.H
FONTOLD.C
HAL.H
HOTSPOT.C

Lines 128,128 - StripePencil() - naked constants 2, 7, 32 need comments.

HOTSPOT.H
IMAGES.C
IMAGES.H
INTRPT.H
IO.H
LOADER.C
LOADER.H
MAIN.C

Line 334 - AdjustContrast() - multiple exit points.

MAIN.H
MEM.H
MEMPLUS.C
MENUS.C

Lines 191-204 - the role and function of naked constants 1,2,3, and 4, as well as the equations are not immediately obvious. Some comments would be helpful.

MENUS.H
MODEM.C
MODEM.H
PACKOFF.H
PACKON.H
PCPRAGMA.H
PEB.C
PEB.H
PRINT.C

Lines 342-494 - cprint() - function is 152 lines long.

PRINT.H
REPORT.C

Line 105 - PullPEBByte() - multiple exit points.

REPORT.H
ROTATION.C
ROTATION.H
RTC.C

Line 131 - SecondsInYear() - multiple exit points.

RTC.H
SELECT.C

SELECT.H
SERIAL.H
SNIP.C

Line 11 - DrawSymbol() - multiple exit points.

STORAGE.C
STORAGE.H
SUPER.C
SUPER.H
TIMELOG.C
TIMELOG.H
TOUCH.C

Line 145 - ReadBoardmA - naked constant needs comments.

TOUCH.H
UTILS.C

Line 115 - imin() - multiple exit points.
Line 122 - imax() - multiple exit points.
Line 218 - pos() - multiple exit points.

UTILS.H
VERSION.H
VOTETYPE.H
VOTEUNIT.C
VOTEUNIT.H
WIBUFF.C

Line 57 - GetWriteInRecordIndex() - multiple exit points.
Line 163 - GetNextWriteIn() - multiple exit points.

WIBUFF.H
WRITEIN.C
WRITEIN.H

2. General Observations

Version 5.12 was an update of version 4.4 code that provided modifications to allow for execution on different platforms, reformatting of parameters passed to functions, and various other things.

The implementation of memcpyforward() and memcpyreverse() strategies also give the impression of attention to detail, resulting in safety and robustness.

If() statements as well as else() and case statements often had the executable statement on the same line as the condition. E.g.

```
        if ( condition ) statement;
        else statement;
```

as opposed to

```
        if (condition)
                statement;
        else
                statement;
```

as well as

```
        case X: statement; break;
```

instead of

```
        case X:
                statement;
                break;
```

This was a minor hindrance to readability, as the indentation strategy was not consistant across the code body. This inconsistancy also led to some anomalies and violations being initially overlooked.

3. 5.12 Final Assessment Statements

Release 5.12 of the Votronic code has been reviewed. The code from ESS continues to be well-written, and exceptionally readable and modular. Some functions exceeded 120 lines in length, but the number of long functions was far beneath the number allowed by the FEC guidelines. Changing the indentation of single-statement if() and else() conditionals as suggested above is encouraged, but not required.

There were more than a dozen instances of multiple exit points (specifically prohibited by the FEC guidelines), none of which was due to a failure in a system call. Each could easily be rewritten with a single-exit-point implementation in a minimal amount of time to be compliant.

It is recommended that after the issues of multiple exit points and uncommented constants are addressed this code be considered compliant with the FEC guidelines for coding practices as set forth within the FEC Performance and Test Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems.

4. Revision 6.1.0 Assessment Statements

All release 6.1.0 files were compared with their version 5.12 counterparts; any differences were investigated with a view of FEC compilance as well as how 5.12 issues have been addressed. The newer files and changed files are listed below.

5. Revision 6.1.0 Source File Specific Notes

BALLOT.C

Lines 675,681,685,686,688,692,693,695,698,703 - PrepareBallot() - naked constants 8,28,24,32,4,25,31,3,48.

Lines 419-598 - FitContestDataOnPages() - function exceeds 120 lines.

Line 801 - CompressPage() - Function has been IFDEFed out as suggested, though if the IFDEF is defined, a multiple exit point situation exists.

Line 843 - DisplayPage() - line is 175 columns wide, a readability issue.

Line 1081 - SetVote() - multiple exit points. - unaddressed from previous review.

Line 1142 - ToggleVote - multiple exit points - unaddressed issue from previous review.

Lines 1269-1366 - PrintFormatedString() - nearly 100-line function has zero informative comments.

Lines 1368-1471 - ReVoteContest() - ~100-line function has zero comments.

Lines 1385,1388-1390,1394-1396,1428- ReVoteContest() - naked constants 3,90,10,80,75,25,50,10.

Lines 1572,1573,1575,1582,1584-1586,1591-1595,1601 - DisplaySummaryInfo() - naked constants 3,4,5,8,28,50,8,39.

Line 1725 - VotePage() - line is 177 columns wide, a readability issue.

Lines 1882-1887 - CastBallot() - code commented out but left in. Will this be coming back?

Lines 2331-2332 - ChooseLanguage() - constants needs comments explaining their values.

Lines 2453-2623 - ChooseLanguageADA() - function exceeds 120 lines with zero informative comments. While this uncommented code is more self-explanatory than other uncommented code some comments would be helpful explaning sections.

Lines 2458-2459 - ChooseLanguageADA() - constants needs comments explaining their values.

Lines 2641-2642 - PracticeRace() - constants needs comments explaining their values.

Line 2838 - playSystemWave() - multiple exit points.

Line 2860 - playNumber() - multiple exit points.

Line 2868 - playAlpha() - multiple exit points.

Lines 3012-3187 - VoteADABallot() - function exceeds 120 lines with zero comments. While this uncommented code is more self-explanatory than other uncommented code some comments would be helpful explaning sections. For example, what is ti? di?

BALLOT.H

BASETYPE.H

Lines 33-85 - can it be guaranteed that the compiler will order the bit variables and other structure members as written?

BINDEF.C

Lines 169,172 - ASCIIToInt() - naked constants 47,58,48. Changing to “>= ‘0’ && <= ‘9’” and “- ‘0’” would increase readability.

Line 209 - IntToASCII() - naked constant 48. Changing to “+ ‘0’ “ would increase readability.

Curiously, functions ASCIIToInt() and IntToASCII() are never called in this code.

BINDEF.H

BOOTIO.ASM

BOOTIO.H

BOOTLOAD.C

Lines 21-209 - all the functions in this file have a total of only two comments.

Lines 28,32,33,37,39,40 - FlashStatus() - naked constants 0x0064, 0x0020, and 0x0044.

Line 43 - FlashStatus() - multiple exit points.

Lines 61,64,65,72,73,74,79 - WriteBuffer() - naked constants 0x3c00000,0x0555,0x02aa,0x00aa,0x0055, and 0x00a0.

Lines 96,98,99,101,108,110-114,121,123,125,127 - EraseSector() - naked constants 0x3c00000, 0x0555, 0x02aa, 31, 0x00aa, 0x0055, 0x0080, 0x00aa, 0x04000, 0x05000, 0x06000, 0x0030, 25000.

Line 144 - SectorCopy() - naked constant 0x3c00000.

Lines 158,160 - SectorCompare() - naked constant 0x3c00000.

Lines 189,190,194,195,200,201 - copyBIOs() - naked constants 16,24,28.

Line 206 - copyBIOs() - is this while(1) supposed to be exited with an interrupt or just stay here forever? Needs comment specifically to explain this.

BOOTLOAD.H

CENTRAL.C

Line 131 - FileDownloadToCommon() - 128K variable allocated on stack. Uncommonly big.

Lines 434-575 - ServiceMenu() - function exceeds 120 lines.

CLOCK.C

COLLECT.C

Line 114 - ConfigureCountersForBallot() - a[b[c(d[e])]] is rather complex. This should be simplified for readability.

Line 243 - AddTerminalVotes() - Could use a comment explaining naked constant 4.

Line 287 - AppendWriteInsToPEB() - Could use a comment explaining naked constant 4.

COLOR.H

CONFIG.C

Lines 157-161 - GetConfigurationFromFlash() - lines commented out but but left in. Why? Is this code coming back?

CONFIG.H

CPFLASH.C

Lines 35,36,45-55 - some compilers would object to the absence of semicolons at the end of these prototypes.

CPFLASH.H
DATALINK.C

Line 502 - PEBEraseSector() - 55-fold decrease in time with no explanation as to why such a significant decrease is legal.

DATALINK.H

Line 30 - comment does not match #define in line 31.

DEBUG.H
DISPLAY.C
DISPLAY.H
EVENTS.C
EVENTS.H

FACTORY.C

Line 325 - storeBurnin() - naked constant 65519.
Line 380 - getBurnin() - naked constant 65519.
Lines 572-645 - getresultstring() - no comments in function - not immediately obvious what is going on, or how the code achieves the desired implementation.
Lines 678,679 - writeBurninfile() - even when file can not be created, the rest of the function continues to attempt to write to the file and close the file descriptor. Rest of function should be inside an else{} for the situation when po_open() is successful.
Lines 788,798,801,804,807,810,816,819 - testMainRam() - naked constants 0x000100000, 0x000100001.
Lines 866,869,872,875 - testRAMDisplay() - multiple exit points.
Lines 887-1082 - TestFlash() - function exceeds 120 lines.
Lines 911,930,942,948,961,980,992,998,1026,1038,1044,1057,1069,1075 - TestFlash() - multiple exit points.
Line 1088 - TouchScreenTest() - multiple exit points.
Lines 1095-1252 - FactoryTest() - function exceeds 120 lines.
Lines 1126-1131 - FactoryTest() - multiple executable statements on one line, specifically prohibited by FEC guidelines.
Lines 1254-1542 - firsttimetest() - function exceeds 240 lines, a length specifically prohibited by the FEC guidelines.
Lines 1266,1271,1274,1277,1280,1283,1288,1291 - firsttimetest() - naked constants 0x000100000, 0x000100001.
Lines 1422-1457 - firsttimetest() - entire function of > 200 lines has only six comments, though most of the function is coded in such a way to be self-explanatory and innate. This section, however, is not as obvious, and could use some more commenting.

FACTORY.H
FLASH.C

Lines 96,99,100,104,106,107 - WaitStatus() - naked constants 0x0064, 0x0044, 0x0020.
Line 110 - WaitStatus() - multiple exit points.
Lines 227,250,255,257,259 - FlashEraseSector() - naked constants 31, 0x04000, 0x05000, 0x06000.
Line 565 - FlashCopyCodeToU11() - multiple exit points - unaddressed from previous review.
Line 583 - FlashCopyCodeToU11() - naked constant 65536*2-4 needs comments. Can it be guaranteed that all compilers have the same "* before -" heiarchy?

Lines 622,631,642 - flashloader() - naked constants 28,32,64500.

FLASH.H
FLATINIT.ASM
FONT.C
HAL.C

Lines 752-879 - ConfigurePhilipsRegisters() - function exceeds 120 lines.

Lines 881-1008 - ConfigureSamsungRegisters() - function exceeds 120 lines.

Line 1161 - DrawColorBox() - naked constant 0x200000.

Line 1192 - DrawColorLine() - naked constant 0x200000.

Lines 1226,1228,1230,1232,1233 - showpalette() - naked constants 16,25,30,20.

Line 1302 - DisplayBitmap2() - naked constant 0x200000.

Lines 1319,1321 - DisplayBitmap2() - lines are 177 and 249 columns wide respectively, a readability issue.

Lines 1338,1345-1347,1354,1357,1369,1375,1379 - DisplayBitmap2() - naked constants 0x0080000, 0xea60,120,4000,600.

Line 1274-1392 - DisplayBitmap2() - ~100-line function has zero comments.

Line 1423 - DisplaySystemBitmap() - naked constant 0x200000.

Lines 1428,1430,1433-1435,1463,1485,1488,1508,1512 - DisplaySystemBitmap() - naked constants 0x3c00000,408,0x0080000,4000,600.

Lines 1442,1444 - DisplaySystemBitmap() - lines are 169 and 241 columns wide respectively, a readability issue.

Lines 1394-1525 - DisplaySystemBitmap() - ~100-line function has zero comments.

Line 1532 - DisplaySystemBitmap2() - Function names of significant length only differing by a single additional character on the end (e.g. DisplaySystemBitmap() and DisplaySystemBitmap2()) are frowned upon as a readability issue. Names should be more different.

Lines 1532-1702 - DisplaySystemBitmap2() - function exceeds 120 lines and has zero comments.

Lines 1562,1567,1569,1572-1574,1619 - DisplaySystemBitmap2() - naked constants 0x200000, 0x3c00000,408,8000.

Line 1748 - IIhzlms() - naked constant 0xFF02.

HAL.H
HAL_A.ASM

Line 116 - _HALClearDisplay - 480000 decimal is 75300hex, not 0x1d4c0 (which is 120,000), as indicated in the comment. In any case, more information explaining the meaning of 480000 should be included in the comment, as 480000 is a naked constant here.

HOTSPOT.C

Lines 132,133 - StripePencil() - naked constants 2, 7, 32 need comments - unaddressed from previous review.

HOTSPOT.H
IMAGES.C

Line 147 - ShowVoteImage() - Could use a comment explaining naked constant 4.

Lines 179,181 - PostVoteImage() - Could use a comment explaining naked constant 4.

IMAGES.H
INTRFACE.H
INTRPT.ASM

Line 121 - StackLeft - naked constant 5000.

Lines 170-190 - _ClockInterrupt - these lines are commented out. Why are they left in? Are they coming back?
Lines 267-290 - _Serial1Interrupt - function is mostly uncommented. This section needs more comments.
Lines 309-331 - _Serial2Interrupt - function is mostly uncommented. This section needs more comments.
Line 343 - _SSIOInterrupt - comments should exist explaining why it is not necessary for this interrupt procedure to preserve its registers.

INTRPT.H
LOADER.C

Line 731, 745 - CompactFlashUpload() - naked constants 24 and 28.
Lines 880-1008 - LoadSystemFiles() - function exceeds 120 lines in length with no comments in the function at all. The function needs comments.
Line 909 - LoadSystemFiles() - naked constant 20.

LOADER.H
MAIN.C

Line 401 - PrintBatteryVoltage() - naked constant 650.
Line 420 - EnforceBatteryVoltage() - naked constant 600.

MAIN.H
MEMPLUS.C
MENUS.C

Lines 196-209 - the role and function of naked constants 1,2,3, and 4, as well as the equations are not immediately obvious. Some comments would be helpful - unaddressed from previous review.

MENUS.H
MODEM.C
MONO.H
OEM.H
P386EX.H
PCDISK.H
PCKERNEL.H
PEB.C
PEB.H
PLATFORM.H
PRINT.C

Lines 345-497 - cprint() - function exceeds 120 lines in length.

PROTINIT.ASM
REALINIT.ASM
REPORT.C

Line 110 - PullPEBByte() - multiple exit points - unaddressed from previous review.
Line 160 - PrintLocalWriteIns() - Could use a comment explaining naked constant 4.

ROTATION.C
RTC.C

Line 129 - SecondsInYear() - multiple exit points - unaddressed from previous review.

Line 526 - RTCTask() - is this line supposed to have a 'break;' at the end of it? Could use a comment explaining this line.
Line 544 - RTCTask() - naked constant 0x31.

RTC.H
SCREEN.H
SDAPI.H
SDCONFIG.H
SDTYPES.H
SELECT.C
SERIAL.C

Lines 369,370 - SSIOInit() - curious how setting two timers with different values produces identical 20ms times, as per comments.

SERIAL.H
SNIP.C

Line 11 - DrawSymbol() - multiple exit points - unaddressed from previous review.

SOUND.C

Lines 146-159 - configSoundDMA() - meaning of naked constants could be explained.
Lines 214-362 - PlayWave() - function exceeds 120 lines and only two lines are commented. Code could use more comments.

SOUND.H
STORAGE.C
STORAGE.H
SUPER.C
TEST.C
TEST.H
TIMELOG.C
TIMERS.C
TIMERS.H
TOUCH.C

Line 118 - ReadBatteryVoltage() - naked constant 30.
Line 120 - ReadBatteryVoltage() - symbolic constant TOUCH_DELAY replaced with naked constant. This is a step backwards in readability.
Line 146 - ReadUnscaledPair() - naked constant 30.
Line 148 - ReadUnscaledPair() - symbolic constant TOUCH_DELAY replaced with naked constant. This is a step backwards in readability.
Lines 527,530-533 - CalibrateTouchScreen() - new code that has been added is commented out. Is this code coming back later?

TOUCH.H

Lines 36, 41-45 - can it be guaranteed that the compiler will order the bit variable members as written?

UTILS.C

Line 39 - AllowPowerDown() - misleading variable name 'yes'. This variable could have a value of 'NO' (as in BALLOT.C, VoteTooManyBallots(), line 2284). As this was not flagged on previous reviews it will not be considered a weighty issue in this review. It is strongly suggested that future versions rename this variable to something less misleading to increase readability.

Line 126 - imin() - multiple exit points - unaddressed from previous review.
Line 133 - imax() - multiple exit points - unaddressed from previous review.
Line 229 - pos() - multiple exit points - unaddressed from previous review.

VERSION.ASM
VOTETYPE.H
VOTEUNIT.C
WIBUFF.C

Line 62 - GetWriteInRecordIndex() - multiple exit points - unaddressed from previous review.
Line 169 - GetNextWriteIn() - multiple exit points - unaddressed from previous review.

WRITEIN.C

6. Revision 6.1.0 Final Assessment Statements

Revision 6.1.0 removed many if() statements that tested for old PEB and Board versions. It is assumed that this code will no longer be required to run on older systems. Also, many debug #ifdefs were removed, as well as many ARG_DEF_ implementations of function parameter declarations.

Many issues were raised with the advent of the new code. Problems with multiple exit points (both new and unaddressed from the previous review), multiple executable statements on one line, and excessive function length (greater than 240 lines) were added, all of which are specifically prohibited by the FEC guidelines.

Many new functions of significant length exist that are sparsely commented, if at all. Some have no comments at all. While most functions are coded in a manner that renders them self-commenting and innately obvious, many uncommented functions were not as immediately readable. Numerous instances of naked constants exist. These raise questions regarding the maintainability of the code. Some lines are excessively long (more than one exceeding 240 columns in width with 8-character tab width) which give rise to readability concerns.

One interrupt routine (_SSIOInterrupt in INTRPT.ASM) did not save its registers on entry nor restore them on exit. This is highly unusual for an interrupt routine, and no comments existed that gave a reasonable explanation as to why this unusual implementation was fine and without consequence.

As many FEC guideline issues mentioned in the previous review were left unaddressed, and considering the new violations of FEC guidelines that now exist, as well as maintainability and readability issues, it is recommended that this code be considered not compliant with the FEC guidelines for coding practices as set forth within the FEC Performance and Test Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems.

7. Revision 6.1.1 Assessment Statements

Release 6.1.1 was received June 4, 2001. All version 6.1.1 files were compared with their version 6.1.0 counterparts; any differences were investigated with a view of FEC compilance as well as how 6.1.0 issues have been addressed. The newer files and changed files are listed below.

8. Revision 6.1.1 Source File Specific Notes

BALLOT.C

Lines 199-208 - KeyCheck() - vendor has fixed a multiple exit point situation that was not even mentioned in the previous review. This is appreciated.
Lines 422-601 - FitContestDataOnPages() - function exceeds 120 lines.

BASETYPE.H
BINDEF.C
BOOTLOAD.C
CENTRAL.C
COLLECT.C
CONFIG.C
DATALINK.C
DATALINK.H
FACTORY.C
FLASH.C

Line 116 - WaitStatus() - different behavior than in release 6.1.0 introduced by fixing multiple exit point - extra call to ReadClock() before returning may set status to 0xffff. While very very unlikely it is still a nonzero possibility. The vendor is encouraged to investigate whether this behavior is acceptable.

FLASH.H
FLASH1.C

This file seems to be simply a backup version of the 6.1.0 flash.c. It is assumed that this file's source code is not included in the production executable.

HAL.C
HAL.H
HAL_A.ASM
HOTSPOT.C
IMAGES.C
INTRPT.ASM
LOADER.C
MAIN.C

Lines 149-152 - CheckStack() - empty function now - was not empty in 6.1.0. Removing a stack check seems to be a step backwards in robustness.

MENUS.C
PRINT.C
REPORT.C
RTC.C
SERIAL.C
SOUND.C
TOUCH.C
TOUCH.H
UTILS.C
UTILS.H
VERSION.ASM
WIBUFF.C

9. Revision 6.1.1 Final Assessment Statements

Release 6.1.1 fixed every single issue raised in the 6.1.0 review, and even one not mentioned. Multiple exit points have been rewritten, and long functions have been broken into smaller, more readable functions. Many very useful comments have been added to previously bare functions, especially when requested to explain constants, increasing the readability significantly. Only one function exists that exceeds 120 lines in length, a number well under the 5% allowed by the FEC guidelines. A great deal of effort has been expended to address every issue raised by the previous review.

Flash1.c is a new file that seems to be an intermediate step between the 6.1.0 flash.c and the fixed 6.1.1 flash.c. It is assumed that flash1.c is not included in the production executables and as such is not subject to review for FEC compliance. Some of the problems in the 6.1.0 flash.c still exist in the 6.1.1 flash1.c, but if this file is not used this is not a problem.

The only issue of concern is the removal of stack checking function code. This function existed in 6.1.0 and seems to be an excellent defensive programming tool. Its removal is curious as the function is still called in several places.

Pending a satisfactory explanation of how the removal of the stack checking function code is inconsequential, and confirmation that flash1.c is not included in the production code, it is recommended that this code be considered compliant with the FEC guidelines for coding practices as set forth within the FEC Performance and Test Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems.

10. Revision 6.1.2 Assessment Statements

Release 6.1.2 was received June 27, 2001. All version 6.1.2 files were compared with their version 6.1.1 counterparts; any differences were investigated with a view of FEC compilance as well as how 6.1.1 issues have been addressed. The newer files and changed files are listed below.

11. Revision 6.1.2 Source File Specific Notes

BALLOT.C
CENTRAL.C
MAIN.C
PEB.C

Lines 396,402,431,437 - RetrieveBallotFromPEB() - why 512? PEB.H line 19 refers to block 512 but it would help if there was a comment nearer the C code itself in future versions.

TOUCH.C
VERSION.ASM

12. Revision 6.1.2 Final Assessment Statements

The changes in this release were few and minor, with no impact on the code's compliance.

It is recommended that this code be considered compliant with the FEC guidelines for coding practices as set forth within the FEC Performance and Test Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems.

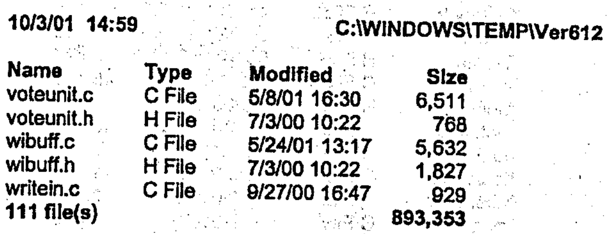
Kevan L. Moore - Contract Engineer

10/3/01 14:59 C:\WINDOWS\TEMP\Ver612

Name	Type	Modified	Size
bootload.h	H File	2/5/01 13:46	629
intrpt.asm	ASM File	5/30/01 16:17	9,200
hal.c	C File	5/30/01 12:02	51,585
bootload.c	C File	5/30/01 11:46	5,954
hal_a.asm	ASM File	5/29/01 15:50	2,049
serial.c	C File	5/29/01 15:18	9,603
sound.c	C File	5/29/01 14:19	11,352
screen.h	H File	5/7/01 12:36	661
clock.c	C File	5/2/01 16:29	2,276
timers.c	C File	5/2/01 16:29	1,800
color.h	H File	4/3/01 14:35	636
realinit.asm	ASM File	4/2/01 09:50	5,295
flatinit.asm	ASM File	3/15/01 17:16	1,109
protinit.asm	ASM File	3/15/01 17:15	1,872
bootio.asm	ASM File	3/7/01 15:03	1,167
bootio.h	H File	3/7/01 15:03	642
timers.h	H File	2/23/01 10:52	1,113
version.asm	ASM File	6/15/01 08:41	501
p386ex.h	H File	12/30/00 16:55	4,824
sdconfig.h	H File	12/23/00 11:05	20,026
platform.h	H File	12/7/00 11:34	456
386ex.inc	INC File	7/3/00 10:20	7,133
buzzer.inc	INC File	7/3/00 10:20	588
mono.h	H File	7/3/00 10:20	583
sdapi.h	H File	4/13/99 09:36	50,580
oem.h	H File	2/19/99 13:40	8,303
pcdisk.h	H File	2/3/99 17:29	21,303
intrface.h	H File	12/17/98 14:49	3,469
pckemel.h	H File	12/5/98 15:36	15,257
sdtypes.h	H File	10/28/98 13:13	2,169
writein.h	H File	7/3/00 10:22	733
ballot.c	C File	6/14/01 13:02	93,694
ballot.h	H File	5/1/01 11:31	6,094
basedef.h	H File	7/3/00 10:22	3,928
basetype.h	H File	5/30/01 11:57	1,781
bindef.c	C File	5/29/01 09:40	5,558
bindef.h	H File	4/29/01 18:12	3,833
central.c	C File	6/14/01 13:50	18,129
central.h	H File	7/3/00 10:22	1,220
clock.h	H File	7/3/00 10:22	1,031
collect.c	C File	5/29/01 10:10	13,251
collect.h	H File	7/3/00 10:22	1,322
common.h	H File	7/3/00 10:22	692
compiler.h	H File	7/3/00 10:22	900
config.c	C File	5/29/01 14:18	9,158
config.h	H File	5/3/01 10:28	3,911
datalink.c	C File	5/29/01 14:18	15,443
datalink.h	H File	5/29/01 14:18	3,203
debug.h	H File	12/20/00 09:02	1,166
display.c	C File	4/30/01 14:42	19,665
display.h	H File	4/5/01 16:58	6,412
events.c	C File	4/29/01 18:04	4,364
events.h	H File	3/16/01 08:58	3,178

10/3/01 14:59 C:\WINDOWS\TEMP\Ver612

Name	Type	Modified	Size
factory.c	C File	5/30/01 10:25	5,475
factory.h	H File	4/29/01 18:08	1,180
flash.c	C File	5/30/01 10:25	16,428
flash.h	H File	5/24/01 13:58	3,891
font.c	C File	9/27/00 16:31	45,419
font.h	H File	7/3/00 10:22	555
fontold.c	C File	7/3/00 10:22	45,350
hal.h	H File	5/30/01 12:02	10,972
hotspot.c	C File	5/29/01 15:50	11,151
hotspot.h	H File	4/19/01 05:12	2,873
images.c	C File	5/29/01 15:50	7,446
images.h	H File	4/26/01 15:50	3,177
intrpt.h	H File	4/29/01 18:16	4,564
io.h	H File	7/3/00 10:22	1,101
loader.c	C File	5/31/01 15:50	28,411
loader.h	H File	3/28/01 13:50	2,037
main.c	C File	6/14/01 13:04	16,447
main.h	H File	4/29/01 20:51	5,059
mem.h	H File	7/3/00 10:22	1,556
memplus.c	C File	2/27/01 15:05	1,456
menus.c	C File	5/29/01 15:18	8,404
menus.h	H File	1/12/01 17:14	2,689
modem.c	C File	4/26/01 16:23	6,380
modem.h	H File	7/3/00 10:22	737
packoff.h	H File	7/3/00 10:22	609
packon.h	H File	7/3/00 10:22	630
pcpragma.h	H File	7/3/00 10:22	999
peb.c	C File	6/15/01 08:57	25,051
peb.h	H File	4/29/01 20:43	7,659
print.c	C File	5/29/01 15:18	8,693
print.h	H File	7/3/00 10:22	2,630
report.c	C File	5/29/01 15:18	18,768
report.h	H File	7/3/00 10:22	1,175
rotation.c	C File	4/29/01 18:04	6,018
rotation.h	H File	7/3/00 11:22	2,619
rtc.c	C File	5/30/01 15:33	16,332
rtc.h	H File	3/8/01 11:41	2,275
select.c	C File	4/24/01 10:13	9,890
select.h	H File	7/3/00 11:22	915
serial.h	H File	2/16/01 19:31	2,930
sound.h	H File	3/4/01 09:23	1,068
storage.c	C File	4/29/01 19:05	5,285
storage.h	H File	4/29/01 19:11	5,254
super.c	C File	4/29/01 19:15	8,649
super.h	H File	7/3/00 11:22	736
timelog.c	C File	9/27/00 16:43	1,529
timelog.h	H File	7/3/00 10:22	724
touch.c	C File	6/14/01 13:02	19,840
touch.h	H File	5/29/01 14:18	3,699
utils.c	C File	5/29/01 14:18	10,693
utils.h	H File	5/29/01 14:18	4,742
version.h	H File	7/3/00 10:22	426
votetype.h	H File	4/30/01 16:06	14,289

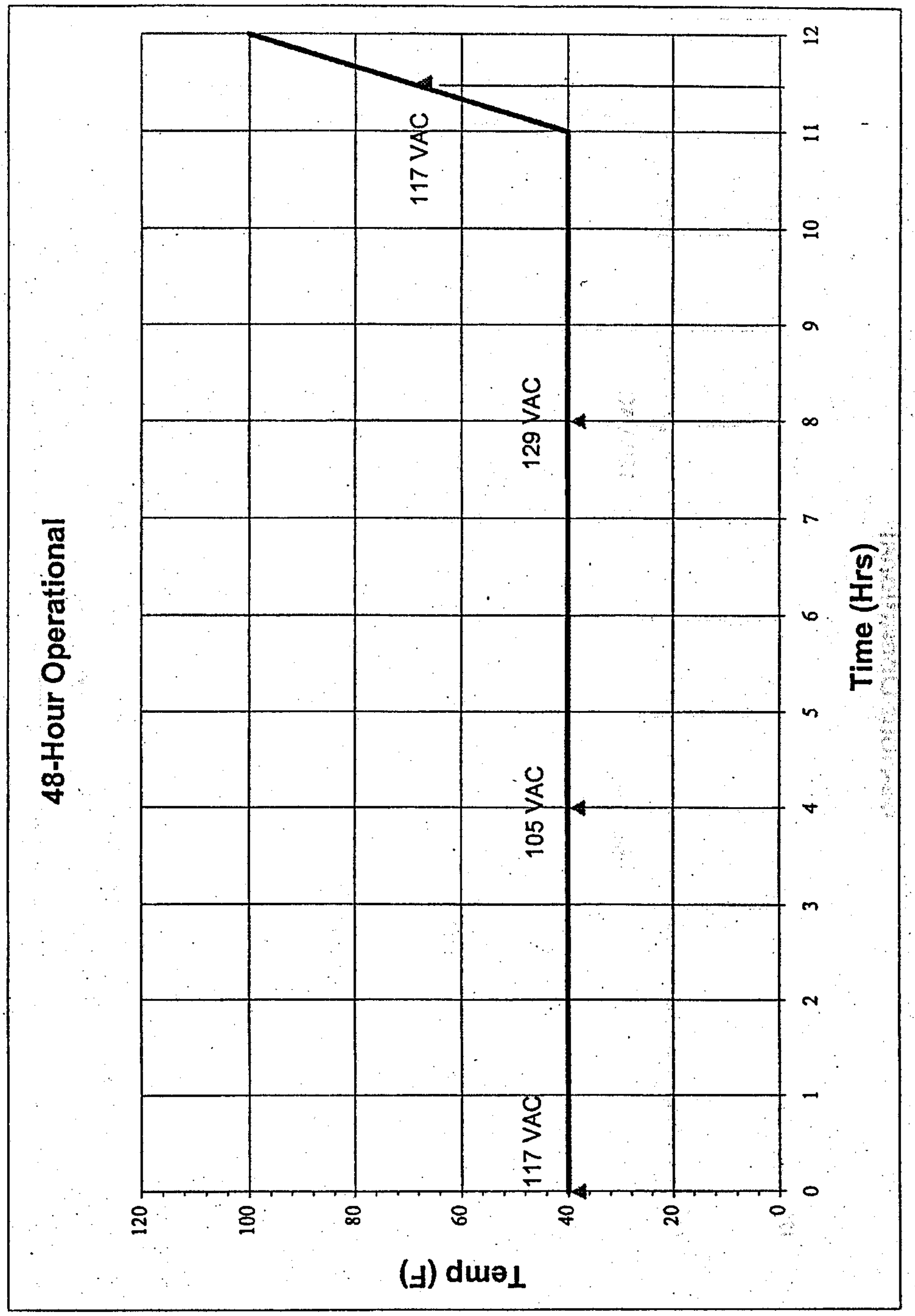


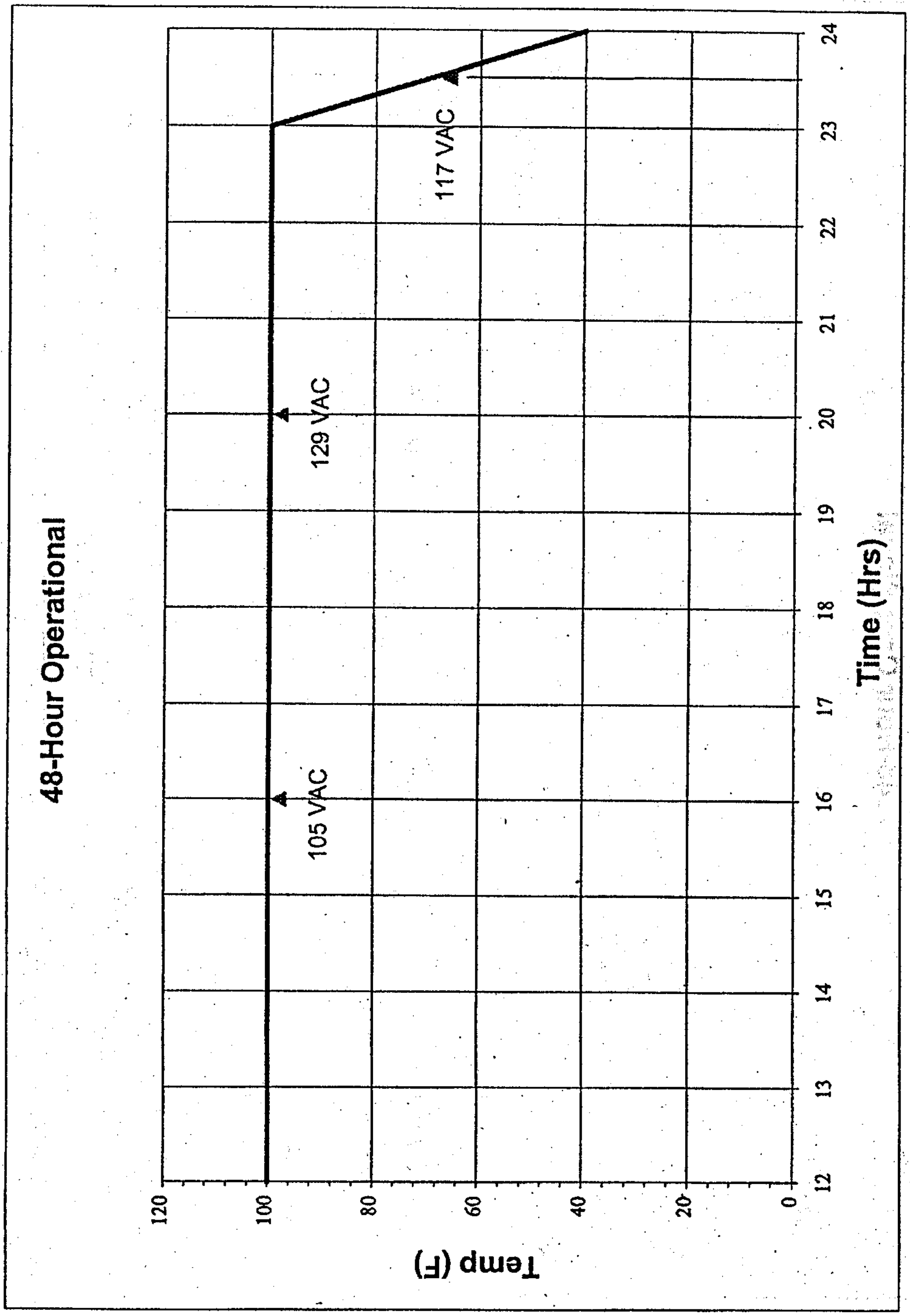
```
10/3/01  14:59                  C:\WINDOWS\TEMP\Ver612

Name          Type      Modified          Size
voteunit.c    C File    5/8/01 16:30      6,511
voteunit.h    H File    7/3/00 10:22        768
wibuff.c      C File    5/24/01 13:17     5,632
wibuff.h      H File    7/3/00 10:22      1,827
writein.c     C File    9/27/00 16:47       929
111 file(s)                             893,353
```

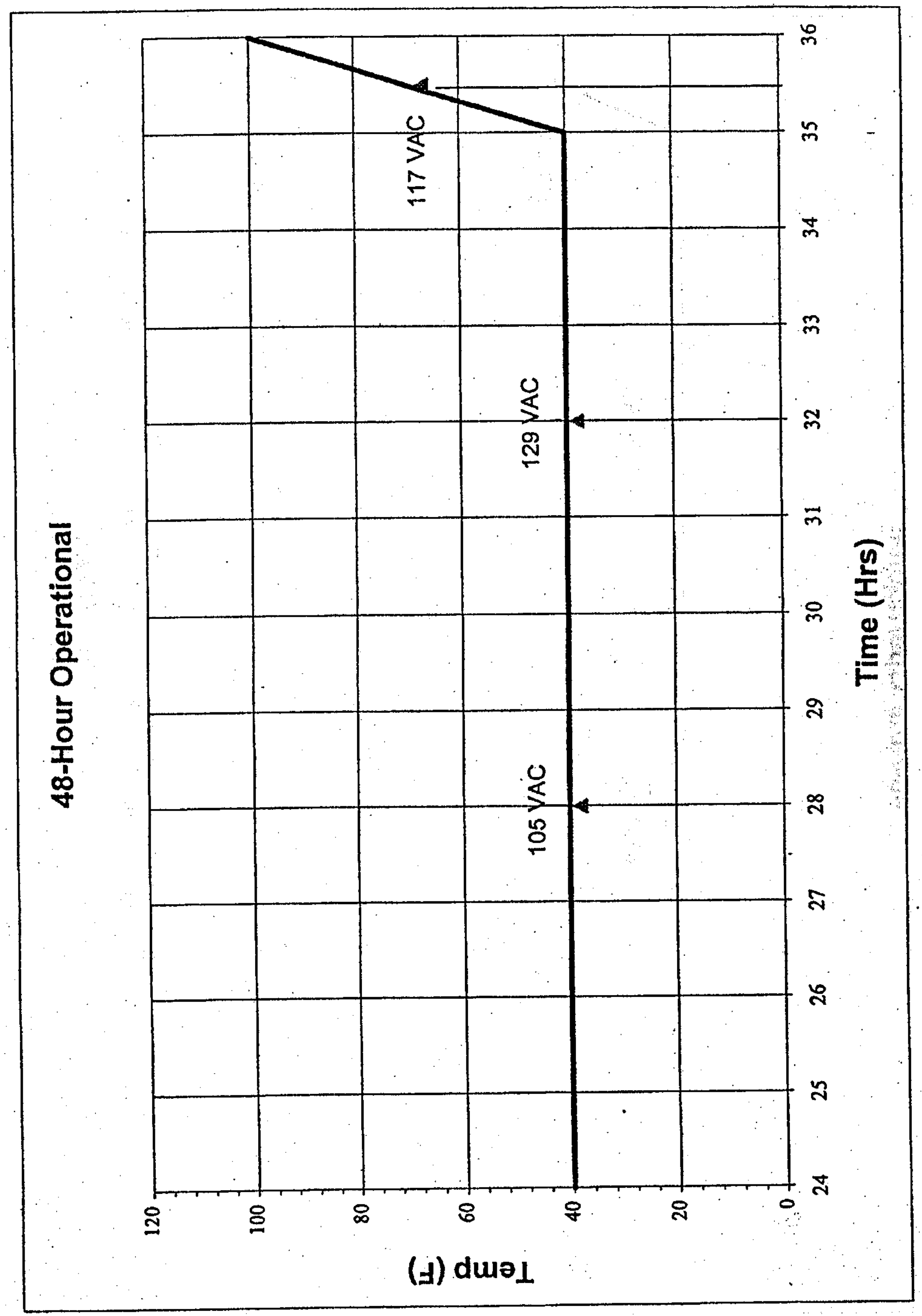
ATTACHMENT C

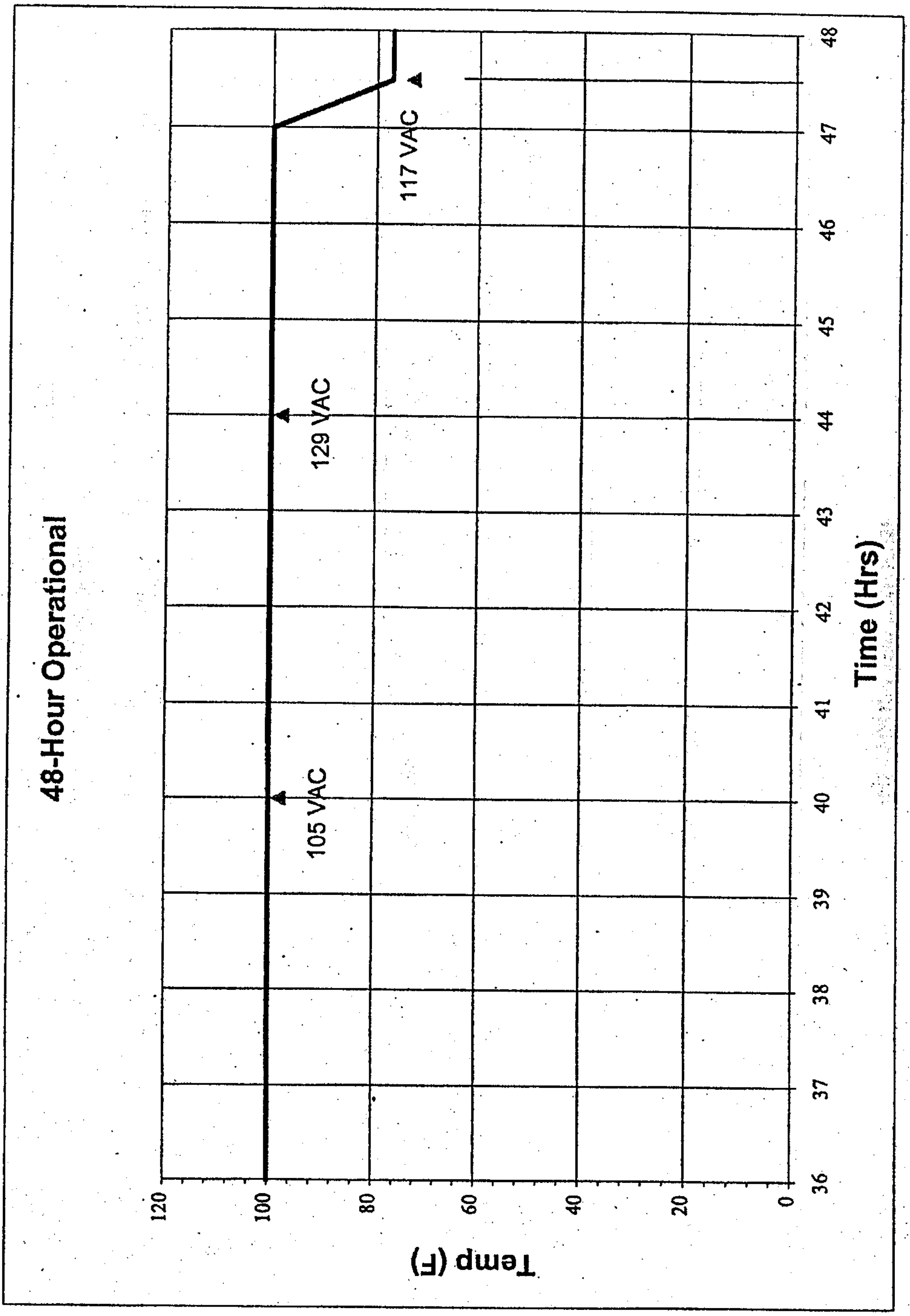
48-HOUR ENVIRONMENTAL OPERATING PROFILE AND THERMAL CIRCULAR CHARTS

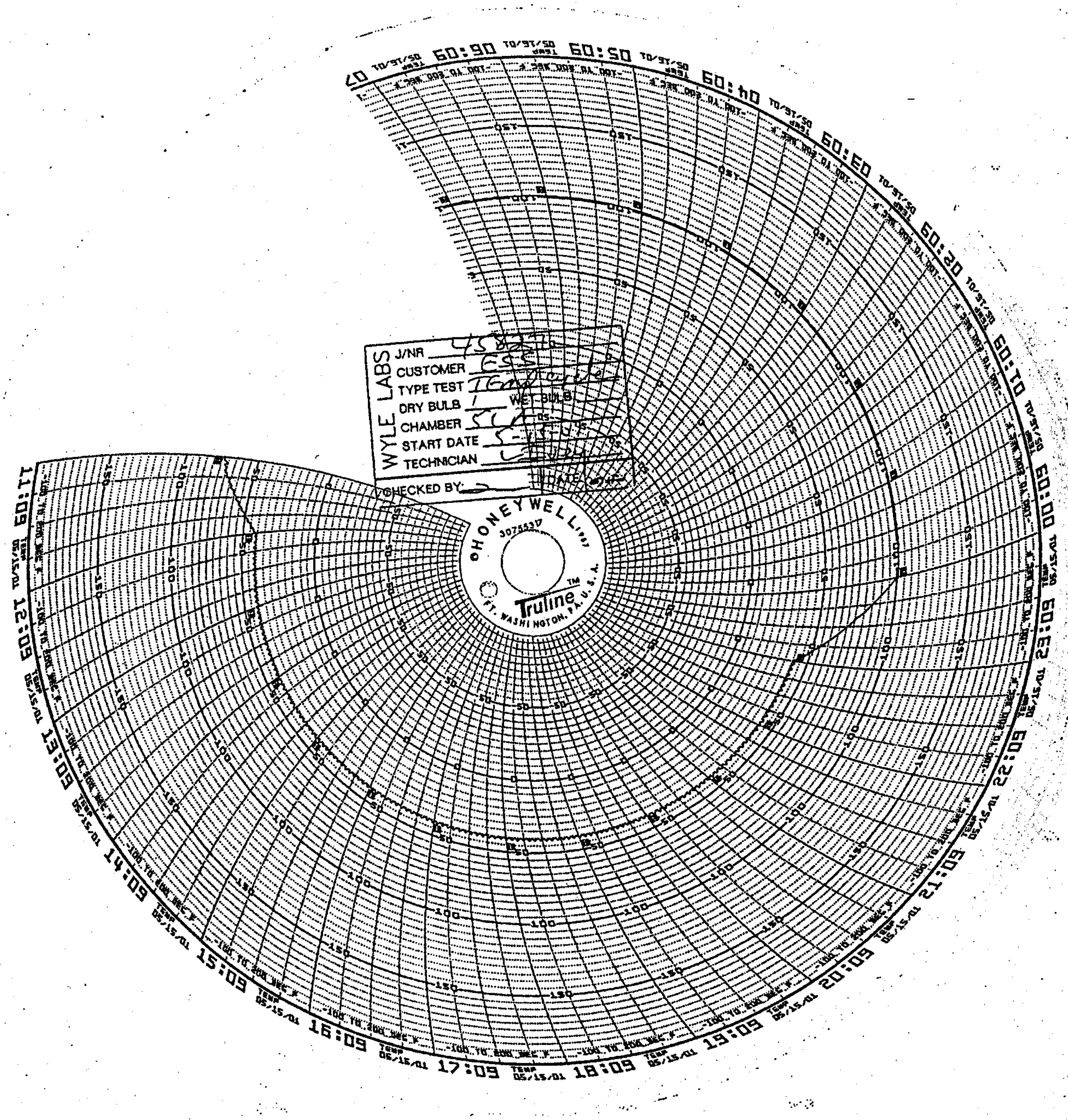
This page intentionally left blank.


48-Hour Operational
Temp (F)
120
100
80
60
40
20
0
Time (Hrs)
0
1
2
3
4
5
6
7
8
9
10
11
12
117 VAC
105 VAC
129 VAC
117 VAC

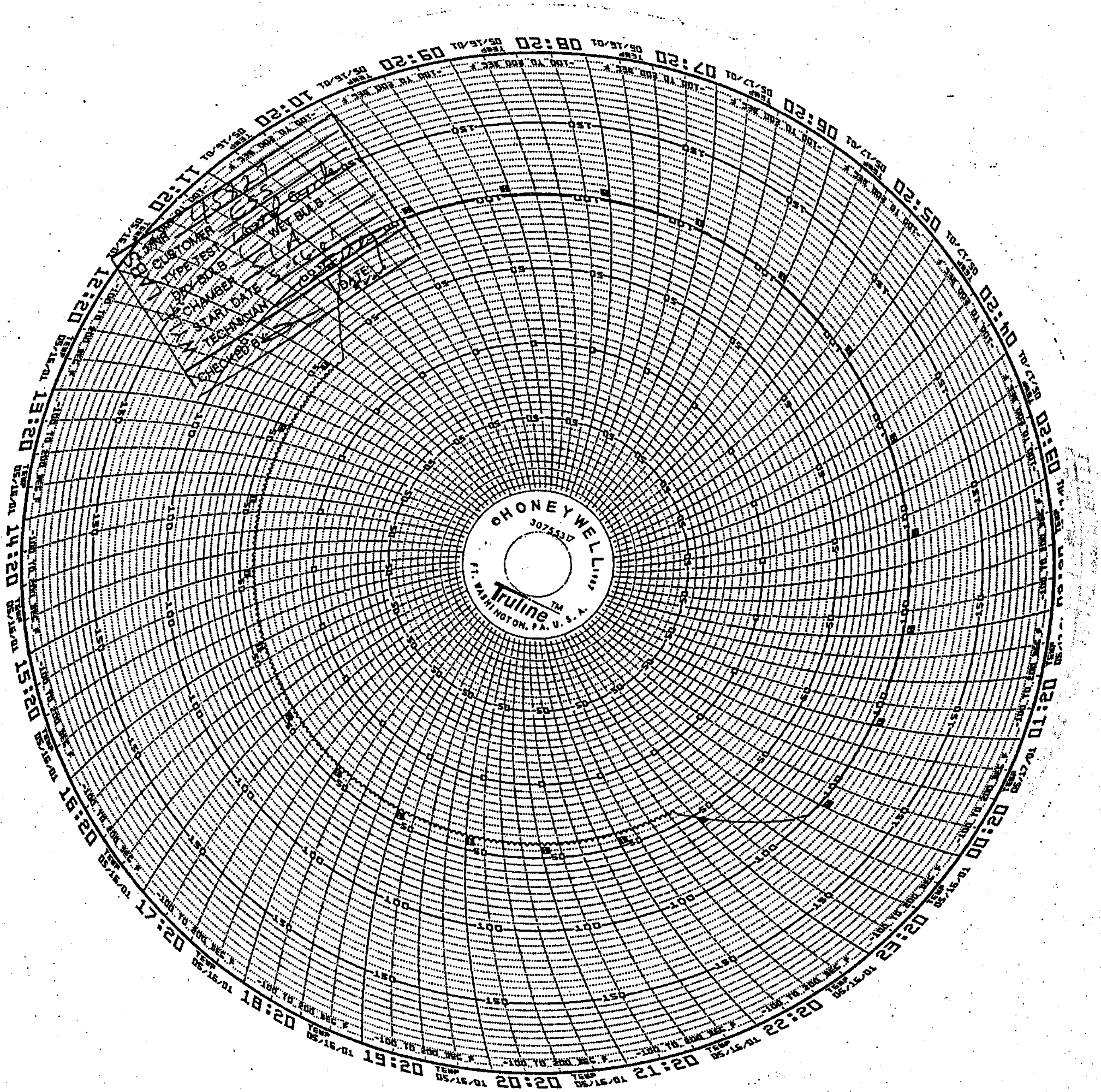

48-Hour Operational
Temp (F)
120
100
80
60
40
20
0
Time (Hrs)
12
13
14
15
16
17
18
19
20
21
22
23
24
105 VAC
129 VAC
117 VAC


48-Hour Operational
Temp (F)
120
100
80
60
40
20
0
Time (Hrs)
24
25
26
27
28
29
30
31
32
33
34
35
36
105 VAC
129 VAC
117 VAC

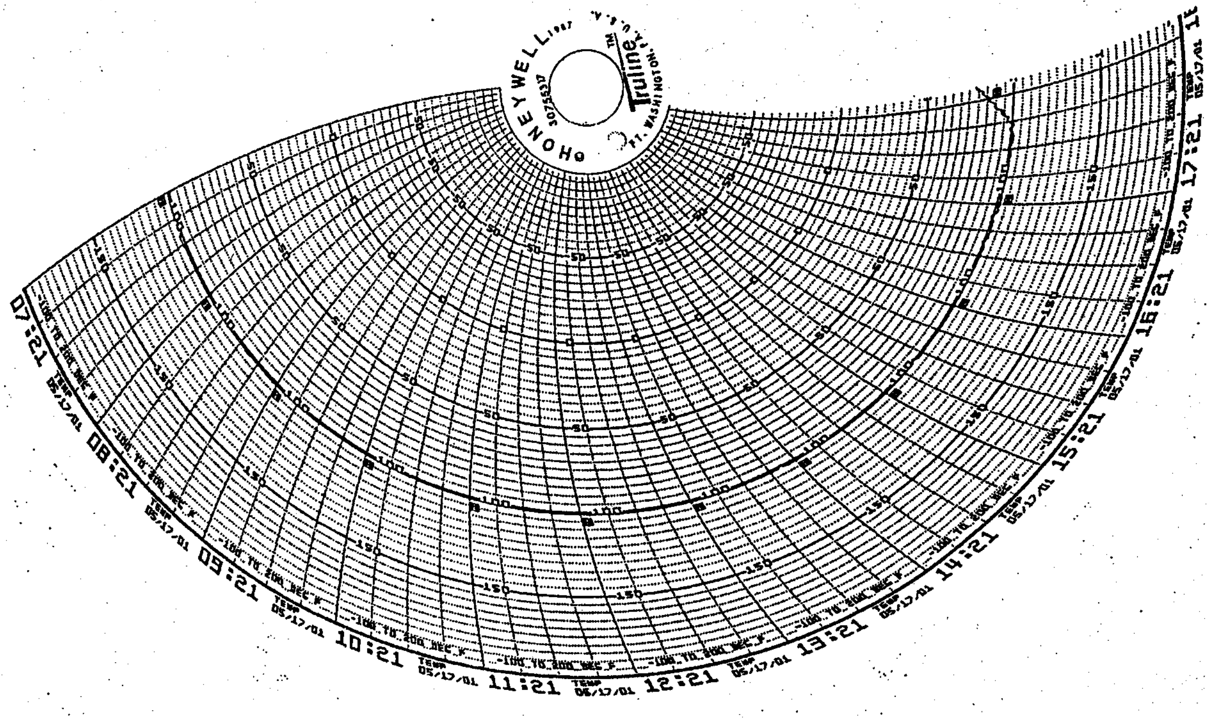

48-Hour Operational
Temp (F)
120
100
80
60
40
20
0
36
37
38
39
40
41
42
43
44
45
46
47
48
Time (Hrs)
105 VAC
129 VAC
117 VAC



WYLE LABS
J/NR 45827
CUSTOMER ESS
TYPE TEST Temp Cycle
DRY BULB 1 WET BULB
CHAMBER SC
START DATE 5-15-01
TECHNICIAN [illegible]
CHECKED BY: [illegible] DATE 5-16


HONEYWELL
30755317
Truline
FT. WASHINGTON, PA. U.S.A.
JOB NO. 45827
CUSTOMER
TYPE TEST
DRY BULB
WET BULB
CHAMBER
START DATE 5-16-01
TECHNICIAN
CHECKED BY
DATE
TEMP
-100 TO 200 DEG F
05/16/01
05/17/01
10:20
11:20
12:20
13:20
14:20
15:20
16:20
17:20
18:20
19:20
20:20
21:20
22:20
23:20
00:20
01:20
02:20
03:20
04:20
05:20
06:20
07:20
08:20
09:20
150
100
50
-50
-100
-150

WYLE LABS
J/NR 45827
CUSTOMER CSS
TYPE TEST Temp cycle
DRY BULB 1 WET BULB
CHAMBER 51-A
START DATE 5-17-01
TECHNICIAN
DATE: 5.17
CHECKED BY



This page intentionally left blank.

ATTACHMENT D

TRANSIT DROP DATA SHEET

This page intentionally left blank.

DATA SHEET

Customer ES&S

Specimen DRE Voting Machine

WYLE LABORATORIES

Part No.	iVotronic 2000	Amb. Temp.	≈ 74°F	Job No.	45827
Spec.	FEC Standard	Photo	–	Report No.	45827-01
Para.	7.3.2.2	Test Med.	--	Start Date	5.21.00 & 8.10.01
S/N	00100063	Specimen Temp.	≈ 74°F		
GSI	No				

Test Title Transit Drop

Criteria: The iVotronic, weighing less than 100 lb. and its largest dimension (fiberboard tranit case) under 36 inches, shall be dropped from a height of 48 inches. A total of 26 drops shall be performed. One drop each shall be performed on each face, edge and corner.

Post-Test Inspection: The iVotronic remained fully functional following the 48-inch drop test.

Criteria: The iVotronic, while packaged in a nylon carrying case, shall be dropped on each corner for a total of eight drops. Each drop shall occur from a height of 12 inches.

Post-Test Inspection: The iVotronic remained fully functional following the 12-inch drop test.

Tested By Date: 10/3/01

Witness - Date: -

Sheet No. 1 of 1

Notice of Anomaly: –

Approved 10.3.01

This page intentionally left blank.