

**DALLAS**  
SEMICONDUCTOR

## Application Note 99 UniqueWare™ Project Setup Manual Revision 2.00B

### SPECIAL FEATURES

- Registered Silicon chips providing globally unique addresses
- Meets the identification requirements of Ethernet, Token Ring, cellular phones, etc.
- EPROM technology for custom programming and quick delivery
- No NRE charge
- Two device options: 512-bit (two pages) and 1024-bit (four pages)
- Dallas Semiconductor handles serialization and does the bookkeeping of tracking the last number used
- Pages are programmed and locked under strict manufacturing control
- Two data structures: one unnamed file or up to three named files (TMEX™ format)
- Unused memory pages can be programmed in the application
- The unique registration number authenticates the chip and tracks the transfer of ownership
- Customer specific Project ID number together with Custom ROM for highest level of product security
- Programmed devices are made available to the owner of the Project ID or their authorized agents only
- Built-in multidrop controller ensures compatibility with other MicroLAN products
- Reduces control, address, data signals and power to a single pin
- Directly connects to a single port pin of a microprocessor and communicates at up to 16.3k bits per second
- Low-cost TO-92 or 8-pin SOIC and 6-pin TSOC surface mount packages as well as F3 and F5 stainless steel microcans
- Tape & Reel option available (not for microcans)
- Reads over a wide voltage range of 2.8V to 6.0V from -40°C to +85°C

IEEE # = 0F 74 A8 36 EE D9  
SERIAL # = 12 5E 70 00 00 54 CF 91



### APPLICATIONS

- After-market protection of products by restricted availability
- Node-ID for network cards (IEEE-assigned number)
- Wireless Phone ID plus battery information
- Electronic product identification label with serialization

### I. INTRODUCTION

UniqueWare is a convenient and affordable way to electronically label items with a unique serialization and to optionally add some text information around the serialization field.

There are several options that allow you to customize what is programmed into UniqueWare devices. This manual will help you when filling in the screens and explain the effect of your entries to take full advantage of all the options. The data you enter will be used directly as a master to program the UniqueWare devices. Therefore it is important that you understand how data is used and what you can expect to find programmed in your UniqueWare devices.

**II. DEVICE DATA STRUCTURE**

**A. Structure Of ROM Section**

As with other Dallas Semiconductor Automatic Identification Products, UniqueWare devices have a ROM Registration Number (Table 1) that serves several purposes:

1. it identifies the logical behavior of the device by its family code
2. it stores the device's unique serial number
3. it validates that a device is genuine UniqueWare. Only UniqueWare Devices have the special identifier code 5E7. Other devices have a different code stored in this location.
4. it provides an 8-bit Cyclic Redundancy Check to verify the integrity of the complete ROM section
5. it may act as node address to singulate and communicate with one device if several Automatic Identification devices are connected in parallel as a 1-Wire™ network.

**UNIQUEWARE DEVICES ROM SECTION Table 1**

**512-bit UniqueWare Device DS2501–UNW/DS1981U**

64-BIT ROM REGISTRATION NUMBER					
8-BIT CRC	UNIQUEWARE IDENTIFIER			DEVICE SERIAL NUMBER	FAMILY CODE
	5	E	7	MSB (hex format) LSB	91
1 byte	12 bits			36 bits	1 byte

high address

low address

**1024-bit UniqueWare Device DS2502–UNW/DS1982U**

64-BIT ROM REGISTRATION NUMBER					
8-BIT CRC	UNIQUEWARE IDENTIFIER			DEVICE SERIAL NUMBER	FAMILY CODE
	5	E	7	MSB (hex format) LSB	89
1 byte	12 bits			36 bits	1 byte

high address

low address

In addition to the ROM, UniqueWare devices have 64 bytes (512-bit device) or 128 bytes (1024-bit device) of one-time programmable data memory organized as pages of 32 bytes each. These pages are factory programmed with a unique Project ID Number, a customer-specified serialization and additional constant data patterns. The UniqueWare identifier in the ROM section together with a unique Project ID number guarantees the authenticity of original UniqueWare parts that are sold only to the owner of the Project ID or other authorized parties. Once programmed, the protected pages of UniqueWare devices cannot be altered.

**B. Structure Of Data Memory**

This section explains how data within the data memory of UniqueWare Devices is organized. There are basically two alternatives (Table 2):

1. combine all pages into one unnamed file (Default Data Structure) containing UniqueWare data; advantage: up to 57 (121) bytes of user data\* available
2. use one page as a directory and store the UniqueWare data as a named file (TMEX format); advantages of TMEX format:
  - a. up to 24 (80) bytes of user data\* available
  - b. give the UniqueWare data a file name
  - c. use TMEX high level functions to read the UniqueWare file
  - d. store up to two additional files together with the UniqueWare file in the same device (1024-bit device only).

\*User data includes the customer specified serialization field.

Regardless of which data structure is chosen, the format of the UniqueWare file is very similar. With Default Data Structure it always starts at physical address 0000H, with TMEX format it starts at address 0020H. Table 3 shows the general structure of the UniqueWare file in case of Default Data Structure. With TMEX format the UniqueWare file may occupy up to 3 pages (1024-bit device). Each page begins with a length byte and ends with a control byte and a CRC16, leaving up to 28 bytes per page for data. As with Default Data Struc-

ture the Project ID is stored in the beginning of the UniqueWare file. Pages not occupied by the UniqueWare file or device directory are available for programming by the user.

In the case of TMEX format, there will be a device directory starting at address 0000H. Details are shown in Table 4. According to TMEX rules, the most significant bit of the 1-byte file extension will be set to 1 to indicate that the UniqueWare file is write protected.

## DATA ORGANIZATION OPTIONS Table 2

### 512-bit UniqueWare Device

	DEFAULT DATA STRUCTURE	TMEX FORMAT
Page 0	UniqueWare File	Device Directory
Page 1	UniqueWare File (cont.) or blank	UniqueWare File

### 1024-bit UniqueWare Device

	DEFAULT DATA STRUCTURE	TMEX FORMAT			
	Only Case	Case A	Case B	Case C	Case D
Page 0	UniqueWare File	Device Directory			
Page 1	UniqueWare File (cont.) or blank	UniqueWare File			
Page 2	UniqueWare File (cont.) or blank	UniqueWare File (cont.) or blank	UniqueWare File (cont.)	second file	second file
Page 3	UniqueWare File (cont.) or blank	UniqueWare File (cont.) or blank	second file or blank	second file (cont.) or blank	third file

## UNIQUEWARE APPLICATION DATA Table 3

### Default Data Structure

LENGTH	PROJECT ID LSB    MSB	TEXT 1 const. data	SERIALIZATION NUMBER LSB (hex. or BCD) MSB MSB (character format) LSB	TEXT 2 const. data	CRC 16 LSB    MSB
1 byte	4 bytes assigned by DS		up to 121 bytes (no file name) extending over adjacent pages		2 bytes

low address

high address

**UNIQUEWARE DEVICE DIRECTORY (TMEX FORMAT) Table 4**

LENGTH	DEVICE CONTROL FIELD							FILE ENTRY 1			
	AA	00	00	00	00	00	01	Name	Ext.	Address	Length
1 byte	7 bytes							5 bytes		2 bytes	

low address

medium address

continued

FILE ENTRY 2				FILE ENTRY 3				CONTROL	CRC16	
N.	E.	A.	L.	N.	E.	A.	L.	00	LSB	MSB
7 bytes				7 bytes				1 byte	2 bytes	

medium address

high address

File Entries 2 and 3 are only possible with the 1024-bit device.

**III. COMPUTER-AIDED DATA SETUP**

**A. Overview**

Several pieces of information are needed to customize UniqueWare devices and to get them programmed for your application. To simplify the process and to avoid misunderstanding and errors introduced by retyping data from paper forms, Dallas Semiconductor has developed a program that prompts for all relevant information, does error checking and writes the result to a machine readable data carrier. Using this program to specify your special UniqueWare device is called a Project Setup Session.

A session to create a set of UniqueWare data consists of seven steps, each having its own screen:

- Screen 1 Address Section, (your office address, for book-keeping purposes)
- Screen 2 Device specification: 512-bit or 1024-bit device
- Screen 3 UniqueWare File, Global Specification (data structure, file name)
- Screen 4 UniqueWare File, Serialization Field (length, counting style, code)
- Screen 5 UniqueWare File, Text Section (optional text around the Serialization Field)
- Screen 6a Second file specification (if space available)
- Screen 6b Third file specification (if space available)
- Screen 7 Saving Data, (write to an iButton™ or disk file)

You may jump forward or backward between screens by using the "Page Up" and "Page Down" keys of your key-

board. You may reach filled-in fields of a screen by using the cursor keys "up" or "down". To replace data, use the backspace key or cursor "left" to empty the field and then type the new data. A session can be ended at any time. If the screen "Saving Data" is reached, the entered data can be saved for later review, approval and ordering.

**B. Getting Started**

To specify the data of your UniqueWare devices, take the UniqueWare Setup Disk, insert it in the appropriate disk drive of your computer and access it. You may do this from WINDOWS using the File Manager or from DOS using the <drive:> and DIR command. In either case you will see the file listing of this disk. The program to execute is called UNIB.BAT.

UNIB.BAT allows up to two parameters to be specified when starting. The first parameter is a 1-digit number specifying the COM-port that will be used to program or read an iButton with your data. If you don't have a COM-Port Adapter with probe for iButtons, this parameter becomes a don't care. You may specify any value from 1 to 4 that your computer accepts. However, you will not be able to program or read an iButton. You will have to store your data on a disk file instead.

The second parameter is alphanumeric. It becomes important if you would like to revise data that you have already specified in an earlier session. If you are using an iButton as data storage, a capital R will tell the program to read data from an iButton through the COM-port specified in the first parameter. If you are using a disk file to store your data, just specify the full name including the path of that file to load its data for review.

Examples:

**UNIB 2**

starts the project setup software and assigns COM-Port 2 to write to an iButton

**UNIB 2 R**

starts the project setup software and assigns COM-Port 2 to first read data from an iButton and to later write back to an iButton

**UNIB 2 DATA.DAT**

starts the project setup software, assigns COM-Port 2 to write to an iButton and loads data from a file with the name DATA.DAT residing in the same directory as UNIB.BAT for review.

**C. Ending the Program**

The program is finished after data has been written to an iButton or disk file. It will then prompt for your permission to quit. However, there is an easy way to quit any time. The ESCAPE key of your keyboard can ALWAYS be used to end the program. You may be asked if you really

want to quit. Enter a Y to quit, otherwise the program will ignore the ESCAPE and resume work at the current location.

**IV. A COMPLETE SESSION**

After a start without the second parameter, the empty Address Section screen appears. This screen is used to enter the customer address including phone and FAX numbers. This information is required to label the container with the programmed parts and to distinguish it from orders of other customers. It also provides a convenient means to contact you to double-check requirements in the case of extremely unusual data specifications.

One can move from one line to the next using the cursor keys "up" and "down" only after data has been entered. However, fields that are not always applicable, such as Mail Stop, State, FAX, may be left empty. The area code (may also include the country code) is separated from the phone number.

**UNIQUEWARE PROJECT SETUP, SCREEN 1**

```

UniqueWare Project Setup Ver. 2.00

Address Section

Company Name      :
Contact           :
Street Address    :
Mail Stop (optional):
City              :
State (optional)  :
ZIP               :
Country           :
Phone             :
                  Area Code  :
                  Number     :
FAX (optional)   Area Code  :
                  Number     :
                  Number     :

(numeric input including (/)- only)

```

After this screen is filled in, one reaches the device specification screen, either after the FAX number has been entered with <RETURN> or with "Page Down". This screen is required to select one of the two possible UniqueWare devices, DS2501-UNW/DS1981U (512 bits) or DS2502-UNW/DS1982U (1024 bits). Depending on which device is selected the global specification screen looks slightly different. In addition, if the 1024-bit

device is chosen and the UniqueWare file itself is very short, the project-setup program may have one or two more screens (screens 6 and 6a).

After the device type is chosen, either by accepting the 512-bit default device or by overstriking the device type 1 by a 2 and pressing <RETURN> or "Page Down" one reaches screen 3 or 3a, Global Specification.

**UNIQUEWARE PROJECT SETUP, SCREEN 2**

```

UniqueWare Project Setup Ver. 2.00

UniqueWare Device Specification

Select Device Type
    enter 1 for 512-bit device
    enter 2 for 1024-bit device
DEVICE TYPE      :1

Hit <Enter> after entering Device Type to continue.

```

The screen “UniqueWare File, Global Specifications” is required to select between two possible data structures, called TMEX Format and Default Data Structure. For details on data structures see Table 2, Data Organization Options. Depending on the selected device type either screen 3 (512-bit) or screen 3a (1024-bit) will appear. The major difference between these screens is that screen 3a allows preselection of three different sizes for the UniqueWare file if TMEX Format is selected.

With TMEX Format (file size option 1 on screen 3, options 1 through 3 on screen 3a), one may either accept the default name IDNR.0 for the UniqueWare file

(UniqueWare file selection # 1) or specify a different name and extension (UniqueWare file selection # 2). The device directory should be hardware write protected (default).

With Default Data Structure (file size option 2 on screen 3, option 4 on screen 3a) a file name cannot be specified.

After screen 3 or 3a has been reviewed or updated one reaches the next screen with “Page Down” or, if the cursor was positioned in the last applicable field, with <RETURN>.

**UNIQUEWARE PROJECT SETUP, SCREEN 3 (512-bit device)**

```

UniqueWare Project Setup Ver. 2.00

UniqueWare File, Global Specifications

Total File Size (including Serialization Field)
    enter 1 for up to 24 bytes (with file name)
    enter 2 for up to 57 bytes (no file name)
FILE SIZE      :1

Name of UniqueWare File
    enter 1 for default name IDNR.0
    enter 2 to specify desired file name
UNIQUEWARE FILE SELECTION      :1

File Name           :<IDNR>
File Extension (0-126) :<0>
WRITE PROTECT DIRECTORY (Y/N) :Y

```

*only applicable with File Size Option 1*

### UNIQUEWARE PROJECT SETUP, SCREEN 3A (1024-bit device)

```

UniqueWare Project Setup Ver. 2.00

UniqueWare File, Global Specifications

Total File Size (including Serialization Field)
  enter 1 for up to 24 bytes
  enter 2 for 25 to 52 bytes
  enter 3 for 53 to 80 bytes
  enter 4 for up to 121 bytes (no file name)
FILE SIZE      :1

Name of UniqueWare File
  enter 1 for default name IDNR.0
  enter 2 to specify desired file name
UNIQUEWARE FILE SELECTION      :1

File Name           :<IDNR>
File Extension (0-126) :<0>

WRITE PROTECT DIRECTORY (Y/N)  :Y

```

*not applicable with File Size Option 4*

The screen “UniqueWare File, Serialization Field” is the central portion of the whole project setup system. Here the specification of the serialization is done. You may specify how much space (number of bytes) is to be reserved for serialization, if incrementing is to occur on a hexadecimal (as done internally in computers) or a deci-

mal (familiar to humans) basis, if the serialization is to be formatted for printing (direct ASCII character representation) or for efficient internal computation (binary number representation) and the starting value for the serialization. The screen example below shows the default settings for the fields.

### UNIQUEWARE PROJECT SETUP, SCREEN 4

```

UniqueWare Project Setup Ver. 2.00

UniqueWare File, Serialization Field

Number of bytes for serialization (2-24)  :4

Counting Style
  enter 1 for hexadecimal
  enter 2 for decimal
COUNTING STYLE :1

Number Representation
  enter 1 for binary (includes 4-bit BCD or hexadecimal)
  enter 2 for character (8-bit ASCII code, with decimal counting only)
NUMBER REPRESENTATION :1

Serialization starts with number:
$ _____ if a $ appears, the starting number has to be entered in hexadecimal form

```

*limits the maximum number of devices that can be programmed with the same project ID*

*preferred for machine-readable serialization e.g., node addresses*

*best for easily printed, human readable serialization such as product identification labels*

The value you specify for “Number of bytes for serialization” directly determines the highest number and with it the maximum possible number range for serialization. If you have decided on the counting style (either hexade-

cimal or decimal) and the format of number representation (unsigned integer true binary/binary coded decimal BCD or character ASCII) then you can calculate the number of bytes required for the application.

As an example, if you need up to but no more than a million different serial numbers, the maximum number to be represented is 999 999.

- Counting in **decimal style** and using **character representation**, each of these digits takes one byte of storage. The Number of bytes for serialization would be 6 in this case.
- If you opt for **BCD number representation** (which also implies decimal counting) then two digits fit into one byte. You would only need 3 bytes for serialization for the same number range.

- If you use **hexadecimal counting** style (which implies binary number representation) then you can distinguish 256 different numbers with one byte. You would therefore also need 3 bytes for serialization using hexadecimal numbers.

Table 5 shows examples for the maximum serialization number depending on the number of bytes reserved and the code to be used. The maximum supported size for the serialization field is 24 bytes. When using character representation for serialization the serialization field must be at least 4 bytes.

**EXAMPLES FOR THE MAXIMUM SERIAL NUMBER** Table 5

Number of Bytes	2	3	4	5	6
hexadecimal	65.535*10 <sup>3</sup>	16.7772*10 <sup>6</sup>	4.29496*10 <sup>9</sup>	1.09951*10 <sup>12</sup>	281.474*10 <sup>12</sup>
decimal, BCD	9 999	999 999	99 999 999	9 999 999 999	999 999 999 999
decimal, character	(99) N/A	(999) N/A	9 999	99 999	999 999

As the last item on this screen, the starting number for serialization may be specified. If the counting style is hexadecimal, the starting number also needs to be specified in hexadecimal (0,1,2, ... 9, A, B, C, D, E, F). This situation is indicated by a \$ sign at the beginning of the entry field. If the \$ is not shown, data is understood as a decimal number. If not otherwise specified, the first value of the serialization number is 0. With character representation, unused bytes ("leading zeros") are programmed as spaces (code 20H). In the case of subsequent UniqueWare orders, note that it is not possible to choose a starting value that would create overlap in serialization with a previous order.

Usually the application determines the counting style and the type of number representation. Hexadecimal counting is the standard choice to realize node addresses for networks with UniqueWare. The decimal counting with character number representation allows one to easily generate printable as well as machine readable electronic labels similar to the standard manufacturers product identification labels.

After the serialization screen has been completed one reaches the next screen with "Page Down" or, if the cur-

sor was positioned in the last applicable field, with <RETURN>. The screen "UniqueWare File, Text Section" allows you to specify up to two text sections surrounding the serialization field. The screen example below shows the default setting for the fields. Depending on your requirements you may select one option of "text section". Any option other than 0 will prompt for the way you want to enter the text.

The most common way (default) is entering the text directly from the keyboard as if you were running a text editor. To select this way, press RETURN. The software will then accept any characters that your keyboard can generate. The character code corresponds to the ASCII code page selected for your computer. (Verification of your text entry on the byte level is explained in section VI. of this document.) While entering text you may use the RETURN key to split the text into several lines. Numbers entered as text are stored with their most significant digit at the lower physical address. (Numbers such as the project ID or a binary serialization are commonly stored in the reverse direction, i. e. least significant byte at the lower address).



## UNIQUEWARE PROJECT SETUP, SCREEN 5

```

UniqueWare Project Setup Ver. 2.00

UniqueWare File, Text Section

The UniqueWare file may contain up to two text sections.
  enter 0 to omit additional text.
  enter 1 to define one text preceding the serialization field.
  enter 2 to define one text following the serialization field.
  enter 3 to define both text sections.
TEXT SECTION :0

Text consists of characters only? (Y/N) :N/A
                                     if you enter N you may enter any data in hexadecimal format

Text 1  :<none>
      |
      | precedes serialization field
      |
Text 2  :<none>
      |
      | follows serialization field
      |
      | required since RETURN is treated as valid character, displayed as
      |
Hit <PageDown> when finished entering text.

```

If you need to store information in a code other than ASCII, enter N followed by RETURN. Now you may enter any information representing text, data or control in hexadecimal form. Enter a space-character ("blank") as a separator between individual bytes. With every byte entered the physical address location within the UniqueWare device will increment.

If you specify two text sections, be aware that both texts are to be entered in the same way (either character or hexadecimal, not mixed). The program will count the characters or bytes you enter and prevent further input if the limit has been reached. Independent of the way you enter text, you may jump between either text section by using the cursor keys "up" and "down". To make corrections, use the backspace key to reach the character or byte to be replaced and re-enter the trailing data. The insert/overwrite function from the DOS command-line editor is not supported. When ready press the PAGE DOWN key to finish the text entry.

With the 512-bit UniqueWare device data entry is now finished and Screen 7 will appear. The same is true for the 1024-bit device if Default Data Structure was selected on screen 3a. With TMEX Format, however, if File Size Option 1 or 2 was selected on Screen 3a, the program will next display Screen 6. The program will not release unused pages if the actual length of the UniqueWare file is shorter than preselected. All possible combinations of files are shown in Table 2. If, for example, two pages are still available the program assigns them

both to the second file; a third file will be accepted only after the data of the second file has been entered and fits on one page. If the UniqueWare file occupies two pages the size of the second file is limited to one page. After all memory is used up or if one does not specify additional files, Screen 7 will appear.

On Screen 6, Additional File Specification, one basically specifies the name of the additional file and enters data. This screen assumes that one intends to store another file. Answering the question "Store Second File?" with N will transfer to Screen 7. The specification of the file name is done in the same way as on screen 3 or 3a. However, there is no default name for the file. Possible file names consist of up to 4 printable characters. After the file name is entered one reaches the field for the file extension by using the cursor down key or by <RETURN>. The default file extension 0 can be over-typed by any decimal number in the range of 1 to 126 or accepted by again using the cursor or <RETURN>. The program will then check if this file name plus extension is already used for another file in the device. If the file name plus extension is duplicated it will display an error message and prompt for a different file extension. After a valid file name is entered one reaches the section of text entry. This section follows the same scheme as with Screen 5 option 1 to 3. Here also <RETURN> is a valid character displayed by a special symbol. To finish text entry one has to use the cursor down key to reach the write protect field. If the file is marked as write protected the file extension stored in the device will be modified to

a value of 128 higher than the original value. The file will also be hardware write-protected, even if you did not enter any text data.

Screen 6a works in the same way as screen 6, but it applies to a third file. After this screen is finished one reaches Screen 7, Saving Data.

Pressing <RETURN> or "page down" finishes the screen and one either reaches Screen 6a or Screen 7.

## UNIQUEWARE PROJECT SETUP, SCREEN 6

```

UniqueWare Project Setup Ver. 2.00

Additional Files Specification (File 2)

There is space for one more data file within the device.
Do you want to store another file together with the UniqueWare?
Store Second File? (Y/N) :Y

File Name           : _____ a file name has to be specified
File Extension (0-126) :0

Text consists of characters only? (Y/N) :Y
Text :
                                     if you enter N you may enter
                                     data in hexadecimal format

Write Protect File (Y/N) :Y
Use the "cursor down" key when finished entering text.
<RETURN> will be treated as valid character, displayed as

```

## UNIQUEWARE PROJECT SETUP, SCREEN 6A

```

UniqueWare Project Setup Ver. 2.00

Additional Files Specification (File 3)

There is space for one more data file within the device.
Do you want to store another file together with the UniqueWare?
Store Third File? (Y/N) :Y

File Name           :
File Extension (0-126) :0

Text consists of characters only? (Y/N) :Y
Text :
                                     (same rules as with screen 6)

Write Protect File (Y/N) :Y

```

Screen 7, "Saving Data to iButton and/or Disk File", is the last of the UniqueWare Project Setup Screens. It handles the dialog on how to store the generated data. The picture below shows how this screen looks initially.

You have two choices: a) storing the result of the session in an iButton and b) saving the result to a disk file. If your computer is equipped for reading/writing iButtons, you have specified a valid COM port number at the time of starting this program and you have a DS1993 or DS1994 available then you can program the iButton with all your UniqueWare specifications. To do this, just press ENTER and follow the instructions on screen.

After an iButton has been programmed, or if you have selected N before pressing ENTER, you reach the section which specifies the name and path of the disk file to

save the data of the session. The default file name is either IDNR.0 or is identical to the name specified on screen "UniqueWare File, Global Specification". The default path is copied from the drive/directory where the Project Setup Software is started from. You may replace the default path and file name with any other specification within the DOS conventions. If you want to finish this session without saving at all, accept the default file information and specify N at the subsequently reached field or press ESCAPE.

If you would like to review the session before saving, simply use the Page Up and Page Down keys to jump from one screen to another. The arrow up/down keys allow you to reach every applicable field. You may erase and retype any data.

## UNIQUEWARE PROJECT SETUP, SCREEN 7

```

UniqueWare Project Setup Ver. 2.00

Saving Data to iButton and/or Disk File

Data Entry for UniqueWare Project Setup is finished.

Copy Data to iButton DS1993? (Y/N) :Y

Disk File Copy information:
File Name :IDNR.0
Path :B:

Copy Data to Disk? (Y/N) :Y
  
```

*enter N if you are NOT equipped to read/write a Touch Memory*

*DOS file name and path conventions apply*

*it is recommended to save the result of a session*

## V. REVIEWING DATA

After the data generated in a session has been saved either to an iButton or a disk file you can load it into the Project Setup program, check it, modify it and save it again some time later. The examples in the section "Getting Started" in the beginning of this manual show how the program knows if it has to read back data and where to get it from. If you load data from an iButton (second command line parameter is R), the program will prompt you to probe the iButton before the Address Section screen appears.

After data has been read from a device or file it is checked for consistency. A modification, for example, from loading the data file to a word processor and saving it even without intentional changes will be identified by non-matching checksums or other inconsistencies. The data will then be useless and has to be re-entered from scratch. A word processor such as NOTEPAD, for example, replaces a 00 byte with a 20H byte, which represents a space rather than a binary number. This modification is detected by the UniqueWare Project Setup Program and the alterations will not be accepted.

## VI. READING OUTPUT DATA

The format of the first two pages (512-bit device) or four pages (1024-bit device) of the data carrier (disk or iButton) that contains the UniqueWare application data are described in detail in the beginning of this document. The only difference between the data submitted and actual programmed devices is that the Project ID field contains zeros rather than your personal Project ID number. Pages 2 and 3 are not accessible with the 512-bit device. They always read "FF" hex in a programmed DS1993 or disk file.

To get the application data into its intended location and to be able to track products during manufacturing, additional information is required. Beginning at page 4 and extending up to page 15, this information is stored as

Default Data Structure with several embedded sections. Each of these sections is preceded by an Output ID String, directly followed by the associated data and terminated by the ASCII code for RETURN acting as a delimiter. Table 6 gives details.

The additional information is invisible for TMEX. However, it can be displayed with the iButton Utility TOUCH.EXE. When stored in a disk file, each 32-byte page of an iButton corresponds to a 32-byte binary record. To examine a file one can use the function "Display File Contents" that the DOSSHELL provides. One should select the "Display HEX" mode to examine on hexadecimal level. Switching to the "Display ASCII" mode clearly shows the Output ID strings, one per line, with its associated data section.

**OUTPUT ID STRINGS** Table 6

OUTPUT ID STRING: MEANING	SPECIFIED WHERE/ THROUGH	VALID CODES/ MEANING
DV: device & data structure	device specification; global specification/file size	1D/512-bit, Default Data Structure 1E/512-bit, TMEX Format 2D/1024-bit, Default Data Structure 2E/1024-bit, TMEX Format
P0: page 0 control info	(derived by combining several data inputs from different screens)	first character: P/program and protect first character: N/no programming first character: Y/program, no protection second char.: C/display text as character second char.: H/display text in hex. format second char.: (space)/C, H not applicable
P1: page 1 control info	(same as with P0:)	(same as with P0:)
P2: page 2 control info	(same as with P0:)	(same as with P0:)
P3: page 3 control info	(same as with P0:)	(same as with P0:)
SESTA: Ser. Start Address	(various sources)	16-bit binary number/byte address of the begin- ning of the serialization field
LEN: Length of serialization	serialization field/num- ber of bytes	8-bit binary number/number or bytes for serial- ization
CNT: Counting Style	serialization field/count- ing style	HEX/true binary counting (not BCD) DEC/decimal counting (BCD or character)
COD: code for counting	serialization field/num- ber representation	BIN/binary coding (hex. or BCD) ASC/ASCII character coding (decimal only)
START#: Starting number	serialization field/ser. start number	binary number of LEN bytes/serialization start number
CN: company name	address section/com- pany name	ASCII string up to 40 characters
CT: contact person	address section/contact	ASCII string up to 40 characters

**OUTPUT ID STRINGS** Table 6 (cont'd)

OUTPUT ID STRING: MEANING	SPECIFIED WHERE/ THROUGH	VALID CODES/ MEANING
AD: mailing address	address section/street address	ASCII string up to 40 characters
MS: mail stop	address section/mail stop	ASCII string up to 10 characters
CI: city name	address section/city	ASCII string up to 30 characters
ST: state name	address section/state	ASCII string up to 10 characters
ZI: ZIP code	address section/ZIP	ASCII string up to 10 characters
CO: country name	address section/country	ASCII string up to 20 characters
PH: phone number	address section/phone	ASCII string, two section of up to 10 characters each, separated by a space
FX: FAX number	address section/FAX	ASCII string, two section of up to 10 characters each, separated by a space
DC: date code	(computer clock)	ASCII string MMDDYY hh:mm:ss R.rr/date of data creation and program revision

## VII. HAND-CRAFTING UNIQUEWARE PROTOTYPES

If you are equipped for programming Add-Only Memories such as DS1982, DS1985, DS1986 (DS9097E COM port adapter, EPROMPC.EXE) then you have everything you need to make UniqueWare prototype devices. These prototypes differ in two important details from real UniqueWare parts: They have neither the UniqueWare ROM code nor a valid Project ID. Such prototypes therefore cannot provide the desired authentication. However, they allow you to look ahead at what you will get and you can continue with the development of your application software. To make such electrically functional prototypes you also need some unprogrammed DS2502 or DS1982 devices, cable to connect the DS2502 as well as an iButton to the DS9097E and a DS1993 for intermediate data storage. As preparation, set up your UniqueWare data structure and write the result to the DS1993.

Next run the program EPROMPC.EXE and copy the data from the iButton to the DS2502/DS1982. You may then write-protect the DS2502/DS1982 with another function of EPROMPC. Now run the UniqueWare project setup software again, restore the data from the

DS1993 iButton, manually increment the Serialization number and store the result back in the iButton. Now you can again run EPROMPC and program the next device, and so on.

## VIII. SKIPPING NUMBER RANGES

Together with your first shipment of ready-to-use UniqueWare devices you may receive a DS1993 iButton containing all your project data plus Project ID and the first serialization number for your next order. To order the next quantity of UniqueWare devices you have to submit this device together with your order form. This iButton eases keeping track of your UniqueWare information and greatly simplifies order processing.

As long as you need more devices with continuously incrementing serialization you have to do nothing with the device but submit it with your order. However, you might have a reason to skip a certain block of numbers and continue at a much higher number. In this case you have to replace the default number by a new, higher number without changing anything else inside the data structure before you submit the device with your order.

Updating the serialization number requires a COM–port adapter DS9097(E) and an iButton Probe DS9092GT (or equivalent electrical connection) installed at your computer. After your computer is ready, start the UniqueWare Project Setup program to load data from the DS1993 (UNIB <port #> R). You will get the prompt:

```
Probe iButton to read data.
```

If the device you probe has a project ID assigned you will get the following message:

```
PID = xxxxxxxx  
PID field is already set.
```

(xxxxxxx will show your Project ID)

```
To update start number only enter (U)  
To begin new project enter (N):
```

After pressing “U” the following message appears:

```
Length Byte = nn  
Current Start Number: yyyyyy  
Enter new Start Number: <new number>
```

Now enter your new Start Number for serialization, followed by <RETURN>. This will transfer you to the screen “Saving Data”, where you have to copy the updated set of UniqueWare data back to exactly the same DS1993 as you loaded it from. Following these steps ensures that the Project ID assigned to your set of data is not erased.

Should you press “N” instead of “U” the program will transfer to Screen 1 and allow all possible other modifi-

cations with the loaded data, but the Project ID will be erased when writing back to a data carrier. If you have pressed “U” and you decide to update the number later, you can quit the program by hitting the <ESCAPE> key.

## IX. ORDERING INFORMATION

The UniqueWare Project Setup Software as described in this document is available on a 3 1/2” DOS floppy disk. To order a copy of this disk, please call (972) 371–6824, send a Fax to (972) 371–3715 or send an EMAIL to [AutoID.Support@dalsemi.com](mailto:AutoID.Support@dalsemi.com). The software can also be down loaded as self–extracting file `unwsetup.exe` from `ftp://ftp.dalsemi.com/pub/auto_id`.

For details on UniqueWare iButtons please refer to the DS1981U/DS1982U data sheet. Details on UniqueWare Add–Only Memories are found in the DS2501–UNW/DS2502–UNW data sheet. These data sheets are available through the World Wide Web and the Dallas Semiconductor Fax–Back Service, phone (972) 371–4441. They are also included in the Automatic Identification Data Book.

All DS2501–UNW and DS2502–UNW UniqueWare Add–Only Memories can be ordered on Tape & Reel. When ordering Tape & Reel, please be aware that there is a certain yield loss in the process of putting the devices on the tape. For the TO–92 and SOIC package this loss is  $1.0 \pm 0.2\%$ , and  $2.0 \pm 1.0\%$  for the TSOC package. To get full reels, it is a good idea to increase the order size by the expected yield loss and have excess devices (if any) shipped separately. Each reel holds 2000 devices in the TO–92 package or 2500 devices in the SOIC or TSOC package.