Outlook

Test Datasheets to Website

From John Lehman <jlehman@dataforth.com>
Date Fri 3/27/2026 9:35 AM
To Mike Swanson <mike@azcomputerguru.com>
Cc Peter Iliya <pIliya@dataforth.com>

Mike,

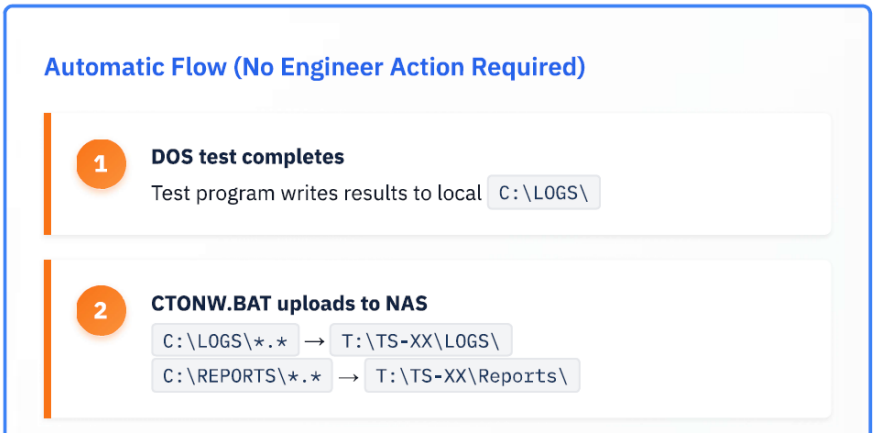
When any test program runs and module passes all tests LOGIT stores data in a binary encoded format and DATASHEETWRITE stores data in a .TXT format

```
Sub LOGMODFILES(STATUS$(), SN$, TSIM!(), OUTCALC!(), OUTMEAS!(), ERROROUT!(), ACCSTAT$(), TSPEC$(), TOTLRESPEC!, TX%)
        'Sub to write the data log, datasheet, and work order status files (and inform the
        'operator of these operations)
        '
        Call WAITFORFINISH  'Pause until "F" key is pressed (so that final test results can be viewed)

        KEY(10) OFF 'Deactivates F10 key

        Cls
        Print Tab(10); "Logging test results."
        Call LOGIT(STATUS$(), SN$, TSIM!(), OUTCALC!(), OUTMEAS!(), ERROROUT!(), ACCSTAT$(), TOTLRESPEC!)      'Logs test data to disk
        Call DATASHEETWRITE(SN$, STATUS$(), TSIM!(), OUTCALC!(), OUTMEAS!(), ERROROUT!(), ACCSTAT$(), TSPEC$(), TX%)
        Print Tab(10); "Writing status to work order status file."
        Call WORKORDERLINE(FAILS%(STATUS$()), SN$)

        KEY(10) ON  'Reactivates F10 key

        Color 11, 0, 0  'Cyan on black background
        Call INTERLUDE

        Call DATASHEETPRINT(SN$, STATUS$())

End Sub
```

Results are appended to a .dat file and stored in the local machine \LOG directory.

```
ub LOGIT(STATUS$(), SN$, TSIM!(), OUTCALC!(), OUTMEAS!(), ERROROUT!(), ACCSTAT$(), TOTLRESPEC!)

    N% = 15
    Do
        LASTCHAR$ = Mid$(SPECS.MODNAME, N%, 1)
        N% = N% - 1
    Loop While LASTCHAR$ = " "

    If UCase$(LASTCHAR$) = "T" Or SPECS.MODNAME = "SCM5B31-1172    " Then
        B% = 1
    Else
        A% = FAILS%(STATUS$())
        B% = 0
    End If

    If A% = 0 Or B% = 1 Then
        Open "C:\ATE\5BLOG\" + RTrim$(Mid$(SPECS.MODNAME, 6, 8)) + ".DAT" For Append As #4

        Open "C:\ATE\LOGPATH.ADR" For Input As #7
            Input #7, LogPath$
        Close #7
        Open LogPath$ + "\5BLOG\" + RTrim$(Mid$(SPECS.MODNAME, 6, 8)) + ".DAT" For Append As #8

        NUMPTS% = 5

        'builds a sequential file.  20 records required for 1 module
        Write #4, SPECS.MODNAME
        Write #8, SPECS.MODNAME

        For INC% = 1 To NUMPTS%
            Write #4, TSIM!(INC%), OUTCALC!(INC%), OUTMEAS!(INC%), ERROROUT!(INC%), ACCSTAT$(INC%)
            Write #8, TSIM!(INC%), OUTCALC!(INC%), OUTMEAS!(INC%), ERROROUT!(INC%), ACCSTAT$(INC%)
        Next

        '"0" place holders are old LOWV! and HIGHV! step response specs.
        Write #4, "0", "0", TOTLRESPEC!
        Write #8, "0", "0", TOTLRESPEC!

        For INC% = 0 To 19 Step 5
            Write #4, STATUS$(INC% + 1), STATUS$(INC% + 2), STATUS$(INC% + 3), STATUS$(INC% + 4), STATUS$(INC% + 5)
            Write #8, STATUS$(INC% + 1), STATUS$(INC% + 2), STATUS$(INC% + 3), STATUS$(INC% + 4), STATUS$(INC% + 5)
        Next

        Write #4, SN$, Date$
        Write #8, SN$, Date$

        Close #4
        Close #8

    End If
```

Your tools copy this to \TEST
When this happens is the source file removed from the test machine?



```
SUB DATASHEETWRITE (SN$, STATUS$(), TSIM!(), OUTCALC!(), OUTMEAS!(), ERROROUT!(), ACCSTAT$(), TSPEC$(), TX%)
    'Sub to write the datasheet file (to be archived to the network and website)
    '
    IF TX% <> 1 THEN EXIT SUB   'Skip writing datasheet file if write to datasheet flag is not "1"

    CALL GETDSFNAME(SN$, DSSNAME$, DSFNAME$)    'Gets datasheet search and file names from serial number

    'Check for module pass or fail
    'IF 0 THEN              'For debug (PWR)
    IF FAILS%(STATUS$()) <> 0 THEN
        'Failing module
        'NOTE:  Currently not creating a file with "Invalid Serial Number" in it, since this type
        '       of "failing" file could overwrite a "passing" datasheet file from later testing
        '       on another tester, depending on the order that the testers are rebooted (and
        '       subsequently when their test files are transferred to the network). Therefore the
        '       the following four lines have been commented out.
'       'OPEN "C:\STAGE\" + SN$ + ".TXT" FOR OUTPUT AS #9
'       OPEN "C:\STAGE\" + DSFNAME$ FOR OUTPUT AS #9
'       PRINT #9, "Invalid Serial Number"
'       CLOSE #9
    ELSE
        'Passing module
        PRINT
        PRINT TAB(10); "Writing datasheet file."
        'OPEN "C:\STAGE\" + SN$ + ".TXT" FOR OUTPUT AS #9
        OPEN "C:\STAGE\" + DSFNAME$ FOR OUTPUT AS #9

        'Get and calculate values.
        SENSOR$ = SPECS.SENTYPE
        MININ! = SPECS.MININ
        MAXIN! = SPECS.MAXIN
        MINOUT! = SPECS.MINOUT
        MAXOUT! = SPECS.MAXOUT
        MAXLINERR! = SPECS.LINEAR                               '%
        MAXACCERR! = SPECS.ACCURACY                             '%
        BANDWIDTH! = SPECS.BANDWIDTH
        LIN% = SPECS.LINEARIZED
        CALOS! = SPECS.OSCALPT / 100            '%
        INRANGE! = SENSOROUT!(SPECS.MAXIN!, 0) - SENSOROUT!(SPECS.MININ!, 0)
        ORANGE! = MAXOUT! - MINOUT!             'Output range (V)

        SNM% = SENSORNUM%

        '------------
        'Write header
        '------------
```

The .TXT format is stored in the local drive \STAGE directory.
Peter has found many files in these directories dating back to Sept 2025.

.TXT files have the filename changed with an alpha character prefix.
I remember doing this but don't remember specifically why.
The datasheet to website program must use this to handle serial numbers of differing lengths and to distinguish from bogus and temporary serial numbers the technicians may use during troubleshooting.

```
Sub GETDSFNAME(SN$, DSSNAME$, DSFNAME$)
        'Sub to create and return the datasheet search name (DSSNAME$) and datasheet file name
        '(DSFNAME) as parameters from the the passed serial number string. The serial number is
        'in the form: "work order number" + "-" "dash number" (for example: "12345-1"). The Sub
        'parses out the work order number from the serial number, checks its validity, and
        'modifies valid six-character work order numbers to create specially coded five-character
        'work order numbers as described below. The modified or unmodified work order numbers then
        'become part of the datasheet search and file names.
        '-------------------------------------------------------------------------------------------
        'The datasheet search and file names produced depend on the parsed work order number as shown
        'below:
        '  1)   If the work order number is 5 characters or less, then the datasheet file name is
        '       "work order #" + "-" + "dash #" + ".TXT" and the search name is "work order #".
        '       For example, if the serial number is "12345-1":
        '       Datasheet file name =   "12345-1.TXT"
        '       Datasheet search name = "12345"
        '  2)   If the work order number is 6 characters with the first two characters from "10"
        '       to "19" then convert the first two characters to "A" from "10" up to "J" for "19".
        '       Then the datasheet file name is "modified work order #" + "-" "dash #" + ".TXT"
        '       and the datasheet search name is "modified work order #".
        '       For example, if the serial number is "123456-1":
        '       Datasheet file name =   "C3456-1.TXT"
        '       Datasheet search name = "C3456"
        ' 3)    If the work order number is invalid (blank, more than six characters, six characters
        '       with the first two characters not a number from "10" to "19" inclusive), then the
        '       datasheet file name is "BAD" + "-" + "dash #" + ".TXT" and the datasheet search name
        '       is "BAD".
        '       For example, if the serial number is "223456-1":
        '       Datasheet file name =   "BAD-1.TXT"
        '       Datasheet search name = "BAD"
        '-------------------------------------------------------------------------------------------
        Call SNPARSE(SN$, WO$, DS$) 'Parse work order number and dash number from serial number
        LWO% = Len(WO$) 'Get length of work order number
        If WO$ = "" Then
                'Work order is blank
                CLS
                CURLOC% = 2
                Color 28, 0, 0  'Flashing light red
                LOCATE CURLOC% + 1, 10: Print "INVALID WORK ORDER NUMBER!"
                Color 12, 0, 0  'Light red
                LOCATE CURLOC% + 2, 10: Print "Work order number is: "; "<blank>"
                LOCATE CURLOC% + 3, 10: Print "Valid datasheet file name cannot be created!"
                LOCATE CURLOC% + 4, 10: Print "Contact Engineering!"
                Color 15, 0, 0  'White on black
                DSFNAME$ = "BAD" + "-" + DS$ + ".TXT"   'Create "bad" datasheet file name.
                DSSNAME$ = "BAD"                        'Create "bad" datasheet search name.
        ElseIf LWO% > 6 Then
                'Work order more than six characters
                CLS
                CURLOC% = 2
                Color 28, 0, 0  'Flashing light red
                LOCATE CURLOC% + 1, 10: Print "INVALID WORK ORDER NUMBER!"
                Color 12, 0, 0  'Light red
                LOCATE CURLOC% + 2, 10: Print "Work order number is more than six characters: "; WO$
                LOCATE CURLOC% + 3, 10: Print "Valid datasheet file name cannot be created!"
                LOCATE CURLOC% + 4, 10: Print "Contact Engineering!"
                Color 15, 0, 0  'White on black
                DSFNAME$ = "BAD" + "-" + DS$ + ".TXT"   'Create "bad" datasheet file name.
                DSSNAME$ = "BAD"                        'Create "bad" datasheet search name.
        ElseIf LWO% = 6 Then
                'Work order is six characters long
                VALLEFT2% = Val(Left$(WO$, 2))  'Get numerical value of left two characters in work order number
                If ((VALLEFT2% < 10) Or (VALLEFT2% > 19)) Then
                        'Value of left two characters are not between 10 to 19 (including the two endpoints)
```

This batch file moves data from local \STAGE to a network drive.
Your process should do the same as this should then be the source for push to website.

U:\ATE\ProdSW\Legacy Test Station Batch Files\BAT\CTONWTXT.BAT - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

TEST5B1E.BAS | TEST5B2E.BAS | NLIBATED.BAS | CTONWTXT.BAT

```
@ECHO OFF
REM PWR: 2013-02-06
ECHO ....................................................
ECHO Archiving text datasheet files to %2 on network...
ECHO CtoNWtxt.bat > C:\ATE\CtoNWtxt.log
ECHO 2013-02-06 Version >> C:\ATE\CtoNWtxt.log
ECHO .................................................... >> C:\ATE\CtoNWtxt.log
ECHO Important! This log file is from the previous >> C:\ATE\CtoNWtxt.log
ECHO            operation since C to Network >> C:\ATE\CtoNWtxt.log
ECHO            archiving is before text file >> C:\ATE\CtoNWtxt.log
ECHO            archiving! >> C:\ATE\CtoNWtxt.log
ECHO .................................................... >> C:\ATE\CtoNWtxt.log
ECHO Copying from C:\STAGE\*.txt to %2  >> C:\ATE\CtoNWtxt.log
IF EXIST C:\STAGE\*.txt GOTO FILEXFER
REM IF ERRORLEVEL=1 ECHO No files were found to transfer
ECHO No files were found to transfer
ECHO No files were found to transfer  >> C:\ATE\CtoNWtxt.log
GOTO END
:FILEXFER
IF NOT EXIST %2\serve.sys GOTO NODIR
copy C:\STAGE\*.txt %2 >> C:\ATE\CtoNWtxt.log
IF ERRORLEVEL=0 ECHO Copy Completed normally
DEL C:\STAGE\*.txt >> C:\ATE\CtoNWtxt.log
ECHO Network archiving of text datasheet files to %2 done!
ECHO ....................................................
GOTO END
:NODIR
ECHO %2\serve.sys not found.  >> C:\ATE\CtoNWtxt.log
ECHO %2\serve.sys not found.  Check Network connection.
:END
ECHO Network archiving of text datasheet files to %2 done!
ECHO ....................................................
```

This .bat runs the one above.

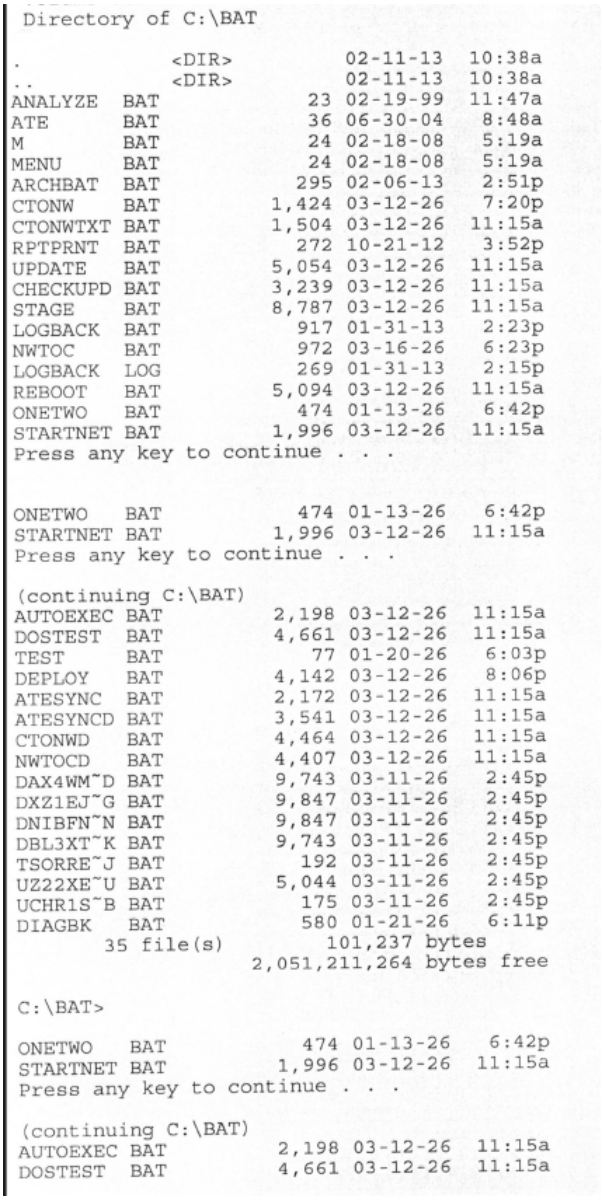
U:\ATE\ProdSW\Legacy Test Station Batch Files\BAT\ARCHBAT.BAT - Notepad++

Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

TEST5B1E.BAS | TEST5B2E.BAS | NLIBATED.BAS | CTONWTXT.BAT | AUTOEXEC.BAT | ARCHBAT.BAT

```
@ECHO OFF
REM PWR 2013-02-06
ECHO ....................................................
ECHO Running network-archive batch files...
ECHO 2013-02-06 VERSION...
Call NWtoC.bat %1
Call CtoNW.bat %1
Call CtoNWtxt.bat %1 %2
ECHO ....................................................
```

Production test stations have many .bat files. We should be able to eliminate many of them.



```
 Directory of C:\BAT

.            <DIR>          02-11-13  10:38a
..           <DIR>          02-11-13  10:38a
ANALYZE  BAT               23 02-19-99  11:47a
ATE      BAT               36 06-30-04   8:48a
M        BAT               24 02-18-08   5:19a
MENU     BAT               24 02-18-08   5:19a
ARCHBAT  BAT              295 02-06-13   2:51p
CTONW    BAT            1,424 03-12-26   7:20p
CTONWTXT BAT            1,504 03-12-26  11:15a
RPTPRNT  BAT              272 10-21-12   3:52p
UPDATE   BAT            5,054 03-12-26  11:15a
CHECKUPD BAT            3,239 03-12-26  11:15a
STAGE    BAT            8,787 03-12-26  11:15a
LOGBACK  BAT              917 01-31-13   2:23p
NWTOC    BAT              972 03-16-26   6:23p
LOGBACK  LOG              269 01-31-13   2:15p
REBOOT   BAT            5,094 03-12-26  11:15a
ONETWO   BAT              474 01-13-26   6:42p
STARTNET BAT            1,996 03-12-26  11:15a
Press any key to continue . . .


ONETWO   BAT              474 01-13-26   6:42p
STARTNET BAT            1,996 03-12-26  11:15a
Press any key to continue . . .

(continuing C:\BAT)
AUTOEXEC BAT            2,198 03-12-26  11:15a
DOSTEST  BAT            4,661 03-12-26  11:15a
TEST     BAT               77 01-20-26   6:03p
DEPLOY   BAT            4,142 03-12-26   8:06p
ATESYNC  BAT            2,172 03-12-26  11:15a
ATESYNCD BAT            3,541 03-12-26  11:15a
CTONWD   BAT            4,464 03-12-26  11:15a
NWTOCD   BAT            4,407 03-12-26  11:15a
DAX4WM~D BAT            9,743 03-11-26   2:45p
DXZ1EJ~G BAT            9,847 03-11-26   2:45p
DNIBFN~N BAT            9,847 03-11-26   2:45p
DBL3XT~K BAT            9,743 03-11-26   2:45p
TSORRE~J BAT              192 03-11-26   2:45p
UZ22XE~U BAT            5,044 03-11-26   2:45p
UCHR1S~B BAT              175 03-11-26   2:45p
DIAGBK   BAT              580 01-21-26   6:11p
       35 file(s)        101,237 bytes
                 2,051,211,264 bytes free

C:\BAT>

ONETWO   BAT              474 01-13-26   6:42p
STARTNET BAT            1,996 03-12-26  11:15a
Press any key to continue . . .

(continuing C:\BAT)
AUTOEXEC BAT            2,198 03-12-26  11:15a
DOSTEST  BAT            4,661 03-12-26  11:15a
```

MENU.BAT is the main port into all test programs for all product lines.
It does not call ARCHBAT.BAT

Current AUTOEXEC.BAT and startup from Station 27

```
@ECHO OFF
REM Dataforth Test Machine Startup - DOS 6.22
REM Version: 4.1 - No IF EXIST checks
REM Deployed: 2026-03-12

SET MACHINE=TS-27
SET PATH=C:\DOS;C:\NET;C:\BAT;C:\BATCH;C:\
PROMPT $P$G
SET TEMP=C:\TEMP
SET TMP=C:\TEMP

CLS
ECHO.
ECHO ========================================
ECHO   Dataforth Test Machine: TS-27
ECHO   AUTOEXEC v4.1 - 2026-03-12
ECHO ========================================
ECHO.


ECHO Starting network client...
ECHO.
CALL C:\STARTNET.BAT

ECHO.
ECHO Network Drives:
ECHO   T: = \\D2TESTNAS\test
ECHO   X: = \\D2TESTNAS\datasheets
ECHO.

ECHO Checking for software updates...
CALL C:\BAT\NWTOC.BAT

ECHO Uploading test data to network...
CALL C:\BAT\CTONW.BAT

ECHO.
ECHO ========================================
ECHO System Ready
ECHO ========================================
ECHO.
CD \ATE
menux
```

Still looking for where/if archbat.bat is called.

John