Appendix F
File Formats

This appendix details the ***formats*** of most ***files*** generated by NUSS. You might need to understand such *file formats* only if your own software applications need to read and process these normally "internal" files that NUSS uses. The appendix also details the ***naming conventions*** used for all files generated by NUSS (see NUSS File Summary Table below).

The **Report** files generated by ***Calibrate Adjustment*** and ***Test*** functions, and the **Playback Secondary Output** files generated by the ***Playback*** feature of ***Run*** functions, are text reports (with **.Txt** file name extensions). These are directly printable and human readable with a simple text editor, thus the *format* of their contents are not described here. However, their *file names* (and the programs that generate them) are listed in a special table at the end of this appendix (see **Section F.3**). All the full report examples generated by various ***Calibration Adjust*** and ***Test*** programs (that once were found in **Chapter 4** and **Chapter 5**) are now included at the end of this appendix in new sections **F.4** and **F.5**, respectively.

The *recorded data files* (with **.Dat** file name extension) generated by the ***Record*** feature of NUSS ***Run*** functions, are fully defined here. Such files contain intermixed *binary* and *ASCII* data. They are normally read only by the ***Playback*** feature of ***Run***. They use the *Binary Access* method of *Visual Basic*, that allows variable-length binary records to be written (Put) and read (Get) sequentially in a file. The format of each of the record types embedded within these files are described fully in this appendix.

Most of the *configuration files* of NUSS (mostly with **.Ini** or **.0** file name extensions) are small *sequential text* files. See **Section F.2** below for descriptions of each these files. These begin with a few *header (comment) lines*, and conclude with a short *body* consisting of *command lines* of the form:

key=value1 [value2]...

Though directly readable, the meanings of the various *keys* and their *parameter values* may require some explanation. Such files probably only need to be processed by NUSS itself, but are detailed in the second part of this appendix. Many of the parameters of these files are also recorded (in abstracted forms) in the *Start* records of *recorded data files* that are described in the first part below. This is particularly true of **Run State** and **Display Set** parameters that are in use by the ***Run*** programs at the time the *data files* are recorded.

Finally, there are several **Comma-Separated Variable** or **“Spreadsheet”** files (with **.csv** file name extensions) that may contain spreadsheet export versions of several of the **Report** files and **Playback Secondary Output** files above. These can be loaded directly into a Spreadsheet program. Some of the *default* naming conventions for NUSS files are listed in the following table:

NUSS File Summary Table

#	purpose	naming convention
1	Data File of Single-Module ***Run***	<*modid*>**-**<*run#*>**.dat**
2	Data File of Coordinated-Group ***Run***	**CG**<*grpid*>**-**<*run#*>**.dat**
3	Test Run # File for Single-Module ***Run***	<*moduleid*>**TRN.ini**
4	Test Run # File for Coord.-Group ***Run***	**CG**<*grpid*>**TRN.ini**
5	Single-Module Memory File for ***Run***	<*modid*>**R.ini**
6	Coordinated-Group Memory File for ***Run***	**CGR.ini**
7	Single-Module **Run State** (0-9) Definition	<*modid*>**RS**<*runstate#*>**.ini**
8	Default **Display Set** of a Run State (0-9)	<*modid*>**RS**<*runstate#*>**.0**
9	Module Group Definition (A-Z)	**G**<*grpletter*>**.ini**
10	Logical Range Number Association	**LRN.ini**
11	Logical Barometer Number Association	**LBN.ini**
12	NUSS General Options memory file(s)	[xxx]**NUSS.ini**
13	Playback Secondary Output Files for ***Run***	<*playbackfilename*><*sid*>**.txt** or **.csv**
14	Secondary ID File for ***Run***	<*playbackfilename*>**SID.ini**
15	Report File for any ***Test*** or ***Cal. Adjust*** form	<*modid*><*repname*>**.txt** or **.csv** or **CG**<*grpid*><*repname*>**.txt** or **.csv**
16	Test/Calibrate Program Tolerance File	**TAccCal.ini**
17	Compensation Accuracy Test Tolerance File	**TComp.ini**
18	Range Codes For Transducers	**RngCodes.txt**
19	Sensor Types for Model 9x46 Modules	**SensType9046.txt**

Key to *field abbreviations* used in above table:

<modid>	=	"*<model#>*-*<serial#>*" (for NetScanner modules)
	or	"<8400>-*<icrs>*" (for System 8400 modules)
<grpid>	=	"**A**" - "**Z**" or "**Star**" (for Group *)
<grpletter>	=	"**A**" - "**Z**"
<run#>	=	**1** .. **32767** (any positive 16-bit integer)
<runstate#>	=	**0** .. **9**
<icrs>	**=**	*I*=interface (0: ethernet, >0: GPIB), *crs*=cluster, rack, slot
<playbackfilename>	=	any ***recorded data file name*** (w.o. **.Dat** extension) played-back by ***Run*** as per items 1 and 2 in table above
<sid>	=	*secondary id* string (e.g., "A".."Z", ..."ZZZ..Z", etc.), used to automatically name **Playback Secondary Output** files.
<repname>	=	a unique Report Name (e.g., TComp) listed in **Section F.3**

WARNING: If you attempt to edit any of the **.Ini** files of NUSS, be sure you retain the exact number of space-separated parameter items as the original. Any parameter enclosed within parentheses ( ) is allowed to have embedded spaces, but the entire parameter must still be space-separated from adjacent parameters.

F.1 Recorded Data Files of ***Run***

The recorded data (**.Dat**) files, generated by the ***Record*** feature of both the *single-module* ***Run*** and ***Group Run*** functions, have basically the same format. In fact, the ***Playback*** feature of ***Group Run*** can read and interpret *both types* of these files equally.

These files are all normally located in the **Dat** subfolder of the NUSS install path. Their naming conventions are shown in items 1 and 2 of the NUSS File Summary Table. The *"group"* version of such files is a true *superset* of the single-module version, and has both *header* records and *data stream* records contributed by *several different* modules (plus a special *synthesized barometer stream*). The *single-module* version uses the same record formats, but with variations contributed by only *one* module (plus the special *synthesized barometer stream*). Other non-module-related *event* records are also added automatically by NUSS (e.g., to record time discontinuities, like pauses during a data recording) or manually by the user (to label some test event). These event records look the same in either type of file.

All records in these *data files* contain variable length binary records, all generated by NUSS itself. The *data streams* (as acquired directly from modules) are not directly written to these files. Instead, NUSS reads the acquired streams into main memory, converting any binary (big endian), decimal, or hexadecimal formatted data directly into its PC-usable binary (little endian) form. It saves the decoded results in special memory buffers (actually "sparse" 3-dimensional arrays, organized by *module*, *data group*, and *channel #*). When data records are written to the *recorded data* files, to represent these acquired streams, all channel data comes directly from the memory resident binary (little endian) data buffers. Thus, they are always written in their most compact binary PC-readable form, regardless of the format of the originally acquired streams. NUSS must also prefix several necessary header fields to all such records, such as: *modid*, *time stamps*, etc. These items are necessary for later use by the ***Playback*** function. They are not found in the original acquired streams, but are necessary for proper presentation of these data in final tabular reports and graphs.

A common 4-byte *binary header* field prefixes *all records* (see example below). This header *identifies* each *type* of record and its *length* in bytes, and facilitates easy ***Playback.*** Using the sequential *Binary Access* method, it is easy to traverse the collection of variable-length binary records (see *Get* and *Put* statements of *Microsoft Visual Basic*).

byte 0	byte 1	bytes 2, 3	byte 4 - n
0	*rec.type*	*record length* in bytes	*other bytes as needed*

The *first byte* (0) of the *common record header* is always *binary zero* (0) to identify a *NUSS- formatted* data file. The *second byte* identifies a particular *type of record* in the file (cataloged below). Note that the *16-bit record length* integer field is recorded in *little endian* (a.k.a. Intel) format (i.e., least-significant byte first). This *common record header* easily distinguishes NUSS data files from a System 8400 data file — which has a similar 4-byte prefix. However, its *record length* bytes are reversed, per *big endian* (a.k.a. Motorola) format, and its first byte is never *zero*.

Generally all *record types* in NUSS *recorded data files* contain mixtures of 1, 2, 4, and 8 byte fields representing simple 8, 16 or 32-bit signed binary integers (and 32-bit IEEE floats), and a special Visual Basic 64-bit Date format (including date and time). These mixtures are organized into larger groupings called *types* (similar to structures in C/C++). All such quantities (larger than a byte) have their bytes ordered *little endian* (Intel) style, to facilitate efficient processing by Windows PC based computers.

Incidentally, the natural command responses and data streams acquired directly from modules (though not recorded directly in these data files) arrive with any datum larger than a byte in a *big-endian* format. Windows recognizes this standard for TCP/IP/UDP headers, and reverses the *endian* of such fields in its Winsock processing code. However, these headers are stripped out before NUSS programs receive them anyway. NUSS programs must convert the remaining data fields (larger than a byte) to little endian format.

F.1.1 Start Records

Every **.Dat** file begins with a **Start** record (RecType=0), containing information describing the *Run State (and Display Set) of each module that contributes stream data to the file*. It does not, however, include any such information for *synthesized barometer streams* (defined later). The following *Visual Basic* declaration begins the *header (first 32-bytes only)* of this *variable length* **Start** record.

```
Type Start_Type       'Typical Record/Playback file's Start Record Header
                            'written @ start of ea. Record/Playback Data file, but..
                            'MAY ALSO BE WRITTEN MID-FILE for ea.module RUN STATE CHANGE
    RecCod As Byte          '0     'Record Code (0=NUSS data)
    RecTyp As Byte          '1     'Record Type (0=Start/Run-State-Change Header)
    RecLen As Integer       '2-3   'Record Length in Bytes
                                   'Above is Common Record Header of all Record Types
    HostTime As Single      '4-7   'Host Time Stamp in Relative Seconds
    RealTime As Date        '8-15  'Real Data/Time
    Group As Byte           '16    'Module Group (0=*, 1-26=(A-Z)=user-def., FF=single)
    CGRunState As Byte      '17    'Coord.Group Run State=0-9 (0xFF=mixed)
    RecRunSeq As Integer    '18-19 'Recorded Data File's Run Sequence # (1-32767, 0)
    NumMods As Integer      '20-21 'Num.contibuting modules (1 if single)
    Spare(1 To 8) As Byte   '22-29 'reserved for expansion & debug convenience
    Ver As Integer          '30-31 'Version of this Type (currently 0)
End Type              'Followed immediately by NumMods appendage(s) of ModInfo_Type
```

In addition to the *common record header* (RecCod, RecTyp, and RecLen) this **Start** record also contains two time stamps. The first (HostTime) is a 32-bit IEEE float containing the *relative system time* (time since the Windows OS started in *seconds*, resolute to about 55 msec. intervals). The second (RealTime) is a 64-bit Calendar Date and Time (Visual Basic *Date* type). Both represent *exactly when* the file is opened (and used as a *base reference* to other *relative* time stamps that are recorded in *data streams*). If the *Run State* of any (or all) contributing module(s) is changed during the time this file is open (and thus being recorded), similar "**Start**" records are also written *mid-file.* These *mid-file* **"Start"** records are written separately for *each* module changing its **Run State** (i.e., NumMods=1), whereas the start-file version is written for *all* modules (i.e., NumMods>1 for ***Group Run*** and NumMods=1 for single-module ***Run***) known to contribute recorded data streams when the data file was opened.

The integer field RecRunSeq has no real purpose above, but contains the same integer used to automatically name the recorded data file (e.g., CGStar-1234 for the 1234th file recorded for modules using Coordinated Group *). The Group field above is always a number, thus 0=Group *, 1..26 = Groups A..Z, and 255 (hex FF) = no group (i.e., a data file recorded for a single-module).

Notes for all Visual Basic data type descriptions: The numbers in the middle column indicate the relative byte indexes of this record in the file. Most types occupy multiples of 16 bytes, making each new type *start* on an even 16 byte line of a hexadecimal file dump (i.e., at relative address XXXXX0). This is indeed a convenience for the programmer, and maybe a waste of space, but also serves to reserve space *in each type* for easy future expansion of features. With this in mind, each type also has a version number (Ver) and various fill bytes given "Spare" or similar names.

However, as already hinted above, this **Start** record is actually much longer than its initial header type, and is also variable length. Any program reading it must deal with this variable length nature, which is a natural feature of the Binary Access method of Visual Basic. First, the above type is followed immediately with a **module-specific** *type* for *each contributing module* that is defined in the *Current Group* being recorded (NumMods > 0 for ***Group Run*** or NumMods = 1 for single-module ***Run***). The following *Visual Basic* declaration defines the first 96-bytes (but beware, it too has other variable length *type* appendages). It also *contains* a nested type.

```
Type ModInfo_Type      'Typical Module Info Hdr.appended to Start_Type NumMods times
    ModID As String * 10      '0-9    'Module's ID ("Model#-Serial#")
    SpareID(1 To 4) As Byte   '10-13  'Spare (e.g., ID additions like "-crs")
    Ver As Integer            '14-15  'Version of this Type (currently 0)
    ModName As String * 16    '16-31  'User's Module Name
    model As Integer          '32-33  'Module's Model #
    serial As Integer         '34-35  'Module's Serial #
    SpareCRS As Integer       '36-37  'Spare for future CRS tag
    NumCh As Integer          '38-39  'Module Max.# Channels
    RunState As Byte          '40     'Run State #: 0=default, 1-9=User-Defined
    DispSet As Byte           '41     'Display State # When File Recorded: 0=default, 1-26
    NumDispSets As Integer    '42-43  'Number of Defined Display sets (# bits set below)
    DispSets As Long          '44-47  'Bit map of Defined Display sets
    RunStateName As String*16 '48-55  'Run State User Name
    Stream(1 To 4) As Str_Type'64-95  'Streams Defined this Run State for module...
End Type               'Followed immediately by NumCh (mod 16) repetitions of Xdcr_Type
```

The last part (bytes 64-95) of each 96-byte ModInfo_Type type, above, defines exactly 4 streams, that are each defined by the following separate type (8-bytes each, total 16 bytes).

```
Type Str_Type          'Typical Stream Info for ea.Module (part of ModInfo_Type)
    BM00 As Long              '0-3    'Bit Map of 'c 00' cmd.: Channels ea.Group
    BM05 As Long              '4-7    'Bit Map of 'c 05' cmd.: Groups (Substreams)
End Type               'NOTE: Synthesized Barometer Stream is 4th stream
```

Each ModInfo_Type *type* has two (2) added variable-length *parts* starting with of a fixed-length *type* defining Barometer/Calibrator associations for each transducer (1 to NumCh) defined in its subject module. Each such Xdcr_Type *type* has the following format.

```
Type Xdcr_Type         'Typical Xducer Info for ea.Ch.of a Module(append to ModInfo_Type)
    LBN As Byte               '0 'Logical Barometer # (0=none, 1-4= LBN)
    LRN As Byte               '1 'Logical Range (Calibrator) # (0=none, 1-8=LRN)
End Type               'Last instance must end on Modulo 16 byte file boundary
```

IMPORTANT: If the variable-length collection of Xdcr_Type types does not completely fill out a modulo 16 byte line of the entire file, then fill bytes (value = 255 or FF hex) are added to the last line until the next modulo 16 byte boundary is reached. This maintains the rule that all data *types* begin on a modulo 16 byte boundary.

So is this the end of the **Start** record? Well no, ModInfo_Type has yet another variable-length appendage for each *Display Set* defined for each module's current *Run State* (1 to NumDispSets). However, for initial releases of NUSS, there can only be one such *Display Set* per module per *Run State* (i.e., NumDispSets = 1, for now). This 48-byte appendage captures the ***Run*** window's basic **Display Set** graphic parameters, including *how many (and which) channels* are displayed on *tabular* display lines. It has the following format:

```
Type Dset_Type              'Typical Display Set Hdr.(another append.to ModInfo_Type)
    DSetName As String*16     '0-15   'User's Display Set Name (unnamed="")
    DSetNum As Integer        '16-17  'Display Set Number (0-26), only 0 currently defined
    NumChan As Integer        '18-19  'Number of Channels Displayed in This Display Set..
                                      '(# bits set in Channels)
    channels As Long          '20-23  'Channels Bit Map
    GraphOn As Byte           '24     'Trend Graph On State (0=False, &HFF=True)
    GraphWrapMode As Byte     '25     'Trend Graph Wrap Mode (0=Clr, &HFF=Erase w.Cursor
    GraphPtSize As Byte       '26     'Trend Graph Plot Symbol Size
    GraphPtsSkipped As Byte   '27     'Trend Graph Symbol Skip Count
    GraphTimeSpan As Single   '28-31  'Trend Graph Time Span (Top to Bottom)
    GraphBkColor As Single    '32-35  'Trend Graph Background Color
    Spare(1 To 12) As Byte    '36-47  'reserved for expansion & debug convenience
End Type
```

This is not the end of the Dset_Type, since a variable length appendage called ChInfo_Type must follow for each tabular channel (1 to NumChan) displayed in the module's current **Display-Set**. These save the display format options of each channel (both EU and "other" tabular fields). Each repetition has the following format:

```
Type ChInfo_Type               'Typical Xducer Info Hdr.(append to ea. Dset_Type)
    EURangeLo As Single        '0-3    'Low End of Displayed Range in EU Bar ®
    EURangeHi As Single        '4-7    'High End of Displayed Range in EU Bar ®
    EUScaler As Single         '8-11   'EU Datum Scaler Used Inside Module (1.0=psi) ©
    EUSpare1 As Single         '12-15  'reserved for expansion

    EUName As String * 6       '16-21  'EU Datum's User Name (default: P1, X1)
    EUType As Byte             '22     'EU Datum's Type (0=Natural, 1=Alternate)
    EUPrec As Byte             '23     'EU Datum's Precision (digits after dec.pt.)
    EUUnits As String * 6      '24-29  'EU Datum's Units
    EUSpare2 As Integer        '30-31  'reserved for expansion
    EUAltScaler As Single      '32-35  'EU Datum's Alternate Scaler (if EUType = 1)
    EUAltUnits As String * 6   '36-41  'EU Datum's Alternate Units (if EUType = 1)
    EUSpare(1 To 6) As Byte    '42-47  'reserved for expansion & debug convenience

    OthName As String * 6      '48-53  'Other Datum's User Name (default: T1, Y1)
    OthType As Byte            '54     'Other Datum's Type (0-2= Temp cnts, V, EU..
                                       'or                10-12=Pressure in cnts, V, EU)
    OthPrec As Byte            '55     'Other Datum's Precision (digits after dec.pt.)
    OthUnits As String * 6     '56-61  'Other Datum's Units
    OthSpare2 As Integer       '62-63  'reserved for expansion
    OthAltScaler As Single     '64-67  'Other Datum's Alternate Scaler (if OthType = 12)
    OthAltUnits As String * 6  '68-73  'Other Datum's Alternate Units (if OthType = 12)
    OthSpare(1 To 6) As Byte   '74-79  'reserved for expansion & debug convenience
End Type
```

This finally ends the variable-length **Start** record at the beginning of each *recorded data file*. Remember that one or more **"Start"** records may also be written **mid-file** if any one module (in the group being recorded) changes its **Run State**. These mid-file records may be smaller than the one at the start of the file (for Group Run only) since it only represents the data from one module. However, if there is a *coordinated* **Run State** change for every module in the Group, it may actually be larger (overall), since each module's **Run State** change is always written separately for *each* module. This adds the initial 32-byte starting header for each module — a header that was shared by *all* modules when it appeared at the beginning of the file.

As you can see (if you analyze the above types completely) ***Playback*** now has the information it needs to properly display every channel datum in every module contributing data — even if one or more modules change their **Run State** (and **Display Set**) mid file.

One final comment on the limits of *item names* in NUSS. Although some of the **.Ini** files that NUSS uses internally would seem to allow *item names* to be truly *variable-length*, it should be clear that the file formats above limit each to some *fixed* field length. Thus all verbose Module, Group, and Run State names (and Event text strings) are limited to 16-characters (not including any surrounding parentheses characters used to shield possible embedded blanks). Data field *names* and their *units* are limited to six (6) characters.

F.1.2 Event Records

Event records (RecType=1 .. 127) may be recorded anywhere needed within the file. The following *type* is used as a fixed (48-byte) header for all these **Event** records.

```
Type Event_Type         'Typical Record/Playback file's Event Header or End Header
    RecCod As Byte              '0     'Record Code (0=all 9xxx modules)
    RecTyp As Byte              '1     'Record Type (1-127=Event types:1=End, 2=TimeJump)
    RecLen As Integer           '2-3   'Record Length in Bytes
                                       'Above is Common Record Header of all Record Types
    HostTime As Single          '4-7    'Host Time Stamp in Seconds
    RealTime As Date            '8-15   'Real Date/Time.
    Group As Byte               '16    'Module Group (0=Group*, 1-26=A-Z. 255=single mod.
    CGRunState As Byte          '17     'Coord.Group Run State # (-1=mixed)
    RecRunSeq As Integer        '18-19 'Record File's Run Sequence #
    NumTxtExt As Byte           '20     'Number of 'text' extension lines (16 bytes each)
    Spare(1 To 9) As Byte       '21-29 'reserved for expansion & debug convenience
    Ver As Integer              '30-31 'Version of this Type (currently 0)
    text As String * 16         '32-47 'Text String Description of Event
    'may be followed by additional 16-byte text lines per NumTxtExt > 0
End Type
```

The header contains 16 bytes for a text comment identifying the event (text), but it may be extended (in 16-byte extensions per NumTxtExt) for event types that require it.
Only RecType=1 (File End Event), RecType=2 (Time Jump Event) and RecType=3 (Test Operator Event) are currently defined. The first two use NumTxtExt=0, and the other occasionally uses NumTxtExt > 0 if the operator’s comment is longer than 16 bytes.

Notice that **Event** records have the same two “true” time stamps that were recorded in the **Start** record(s). These help gauge the true length of recorded “test runs” that could last for some time, and might even span days.

RecType = 1, called the **File End Event**, is only recorded when the user of a ***Run*** form presses the **[Rec]** button a second time to close the file. This event record only serves to record the date and time that the recorded data file ends. The **Start** record serves the similar purpose of capturing the date and time of the file opening.

RecType = 2, called the **Time Jump Event**, is recorded by NUSS whenever any of the following conditions exist:

1. Recording data file is opened, but ***Acquire*** is Off (i.e., there is no immediate data to record). The event indicates why the *first* recorded data stream has a *relative file time stamp* much *later* than the opening of the file (i.e., relative file time = 0). This **Event** record immediately follows the file’s **Start** record.

2. Recording is Paused manually by operator. The event indicates why the *next* recorded data stream has a *relative file time stamp* much *later* than the *previous* recorded data stream. This **Event** record may appear anywhere in the file between **Start** record and **End Event** record.

RecType = 3, called the **Test Operator Event**, is recorded when the user wishes to record various comments during a test run. To do this, the operator of a ***Run*** form clicks the **Opt** *check box* in the **Record Controls** frame. This pops-up the **Record Options** form. If recording is already On or Paused, the form has a *text box* to accept any suitable text message. An **[OK]** button then manually records this text as a **Test Operator Event**. If event messages longer than 16-characters are entered by the operator, then additional 16-byte text lines are written to the event record to accommodate them. However, there is room on the Bottom Status Line of a ***Run*** form to show only the first part of long Event messages (during ***Playback***).

F.1.2 Data Stream Records

Data Stream records (RecType=255, or hex FF) are recorded anywhere in the file, after the *first* **Start** record (and before the **End Event** record) whenever:

(1) **Acquired "live" data streams** are available to be recorded, originating from a single module or the module(s) defined in the current Group; or

(2) it is necessary to record a **Synthesized Barometer Stream** to the file.

The following Data_Type *type* is used as a fixed (48-byte) header for all these variable-length **Data Stream** records. Stream Data Groups, in an IEEE Single Float array (*little endian*), follow this header. They have the same format as the *data* (only) in a naturally acquired module stream (except for reversing the *endian* format).

```
Type Data_Type        'Typical Record/Playback file's Data Stream Record
    RecCod As Byte           '0      'Record Code (0=all 9xxx modules)
    RecTyp As Byte           '1      'Record Type (255=EU Data)
    RecLen As Integer        '2-3    'Record Length in Bytes
                                      'Above is Common Record Header of all Record Types
    RecTime As Single        '4-7    'Record Time Stamp in Seconds Hdr)
    model As Integer         '8-9    'Module's Model #
    serial As Integer        '10-11  'Module's Serial #
    XXXX As Long             '12-15  'Stream's Sequence # (0=prefix)
    VV As Integer            '16-17  'Stream's Valve Status (-1 if none)
    TT As Integer            '18-19  'Stream's Temp/Aux. Status (-1 if none)
    Stream As Byte           '20     'Module's Stream # (1-3=real, 4=special Barometer)
    NumValues As Byte        '21     'Number Data Values in stream (1-180)
    Spare(1 To 8) As Byte    '22-29
    Ver As Integer           '30-31  'Version of this Type (currently 0)
End Type              'Followed by acquired/synthesized datum stream (reconstructed)
```

The *record time stamp* (RecTime) is similar to the other IEEE single float *time stamps* (HostTime) already shown in the **Start** and **Event** records. However, RecTime has the HostTime field (from the **Start** record) *subtracted* from it. Thus, *relative time* is always relative to the time when the *recorded data file* was initially opened (i.e., when the **Start** record was written to the file). Unfortunately, the natural data stream captured in this record has no suitable time stamp, thus this one is written by NUSS at the moment when the TCP data stream arrives in its host computer. This might be *somewhat later* than the real time at which the stream is scanned by the A/D converter inside the module. This time difference might account for any observed "jitter" in data plotted on a graph (like the Trend graph). Adding such a time stamp to the module's emitted data stream would be a preferable future product enhancement, but is impossible until a suitable time-code signal can be easily distributed to each module on the network.

The header also contains the <*model#*> (model) and <*serial#*> (serial) of the module that generated the data stream. Playback uses this to locate the source module's data display later. The header also contains the *Sequence #* (XXXX, similar to scan #) of the original natural stream (1 .. n). This value is zero (0) if the stream was the single Prefix stream.

Notice that this header also has two fixed 16-bit integer fields (VV and TT) that reflect natural (but optional) **Status Data** (Valve Sense Status and Temperature Alarm Status) in a module's natural emitted autonomous stream. The values -1 are assigned to these fields when the module did not have such a status value prefixed to its natural autonomous streams (i.e., they were not specified as *status* prefixes by the definition of your active *Run State,* for module types that can provided them).

A *real* Data_Type header (Stream=1-3) is followed immediately by a **Real Acquired Stream** Data *type* — which is always a variable-length Array (1 To NumValues) of Single IEEE Float. Although such recorded data streams are constructed from the *data* already in memory buffers for the acquired stream, it has the real streams format (as defined in the *Run State* for that stream (Primary, Secondary, or Tertiary) which is currently being used by the source module). The first such datum is the *highest-numbered channel scanned* for the *highest-numbered data-group included* in the stream definition — and the last such datum is the *lowest-numbered channel scanned* for the *lowest-numbered data-group included* in the stream definition. Refer to the Bit Maps defined by the '00' and '05' sub-commands of the low-level 'c' module commands that define all data streams. These bit maps were recorded in the Start record (for each stream or each module) using the Str_Type *type* (fields BM00 and BM05) replicated four times at the end of the ModInfo_Type *type* (see **Start** record type).

A *synthesized* Data_Type header (Stream=4) is followed immediately by a special **Synthesized Barometer Stream** which is occasionally recorded to the file with the same header, except that its NumValues field is always 5. This records five (5) Single IEEE floats, where the first (index 0) is a constant (14.7 psi), and the others (index 1-4) are the "latest readings" from up to four barometers (known by their index positions in this array). The index (1-4) corresponds to the Logical Barometer Number (LBN) used to associate any transducer of a module to a particular Barometer module. This Barometer Stream record, when included in a data file, is found immediately following the Start record, and recorded periodically thereafter while the file is open. If no transducers of any contributing module(s) have any LBN associations assigned, it is not recorded at all. Please note that no Barometer need be defined in the current Group, thus there are no contributions from it in the Group's Run State (and Display Set) parameters recorded in the file's Start record(s). Each "active" Barometer does need to be running in its own (single-module) ***Run*** form, however, in order for it to be contributing data to the **Synthesized Barometer Stream**. It does this by simply depositing its latest pressure datum in a "global" buffer, indexed by its assigned LBN, when it runs. This synthesized stream is written directly from the same buffer.

Both types of *data streams* are padded by 4-byte longs (-1 or hex FFFFFFFF) as necessary until the current 16 byte file line is filled out. This insures that the next file record begins on a modulo 16 byte boundary.

F.2 Configuration Files of NUSS

Most configuration files of NUSS (normally with **.Ini** file name extensions) are located in the **Ini** subfolder (of the NUSS install path). However, two of them are located in the **Dat** subfolder (of the NUSS install path) so that they can be with the recorded data files they support (see next section).

F.2.1 Current Test Run # Files For Automatic File Naming

There are two types of <*various*>**TRN.ini** files which are normally located in the **Dat** folder with the *recorded data files* they support (see items 3 and 4 in the NUSS File Summary Table). One supports the single-module **Run** function's automatic naming of recorded data files, and the other does the same for ***Group Run***. The only difference is the way these files are named. The file content is the same for both, and possibly wins the award for the *simplest file format of all time*! That is, each of these files is a sequential text file with exactly one data line, and that data line contains exactly one formatted decimal integer called the <run#>.

This number is the next <run#> used to automatically name the next recorded data file created by ***Run*** or ***Group Run***. See items 1 and 2 in the NUSS File Summary Table for how it is used in each case. If NUSS finds any of these files deleted for a particular module or module Group, it creates a new one with the number 1 in it. That is the next <run#> used to name a new data file. Whenever such a data file is actually opened, this ***.ini*** file is rewritten with the *next* <*run#*> number in sequence stored in it. This updated value is then used to name the next file, and so on. After it reaches the number 32767 (largest positive 16-bit integer) it wraps to zero (0) for its next value, and then start repeating the initial range thereafter. Hopefully, the user has moved his older data files outside the **Dat** subfolder (of NUSS install path) before such a wrap around occurs. When all files are removed from this subfolder, by normal file maintenance methods, normally using My Computer or Windows Explorer, these <run#> files should also be deleted, so that they begin (at 1) again.

F.2.2 Memory Files for Modules/Groups

There are two types of "historical" Memory (<*moduleid*>**R.ini** and **CGR.ini)** files used by the ***Run*** and ***Group Run*** functions. They are located in the **Ini** subfolder of the NUSS install path (see items 5 and 6 in the NUSS File Summary Table). They are described separately below.

The <*moduleid*>**R.ini** file's content remembers everything about the Size and Location of the "module-specific display form" of ***Run*** and ***Group Run***, but it also remembers other information about its *module* in general, such as the *module name* (ModUserName =**()**, or *empty string*, in this example). It remembers the *recorded data file path* (DatPath) when used with *single-module* ***Run*** only. It remembers the module's "current" **Run State** (and **Display Set**), and the Barometer (LBN) and Calibrator (LRN) associations of each of the module's channels (only 12 for this Model 9021 example). Other scanner modules would have additional LBNLRN= keyed lines for additional channels. Calibrator and Standard modules would not have any of these keyed lines.

```
--- NUSS 'Run Form' Initialization File for Module: 9021-141 () Scanner
--- Module Firmware Ver:   2.05
--- Using:     NUSS Version: 0.1.1
--- Updated: 08-08-2000 at 16:43:38
Size= 3225 6735
Location= 1365 8430
ModUserName= ()
DatPath= (D:\Nuss\Src\Dat\)
RunState= 2
DisplaySet= 0
LBNLRN1= 0 0 0 0
LBNLRN2= 0 0 0 0
LBNLRN3= 0 0 0 0
LBNLRN4= 0 0 0 0
LBNLRN5= 0 0 0 0
LBNLRN6= 0 0 0 0
LBNLRN7= 0 0 0 0
LBNLRN8= 0 0 0 0
LBNLRN9= 0 0 0 0
LBNLRN10= 0 0 0 0
LBNLRN11= 0 0 0 0
LBNLRN12= 0 0 0 0
```

Each LBNLRN= keyed line has four space-separated parameters. The first and second parameters respectively specify the assigned LBN (0=none, or 1-4) and LRN (0=none, or 1-8) of that channel of the module. The third and forth respectively specify the "valve index" to be set during the calibration/test function, and the "valve index" to be set after that function is completed. Valve indexes are 0=Manual(prompted), 1=Cal, 2=Run. When specific Cal or Run valves are set for modules WITHOUT any valves, it means that the program should assume external pneumatic plumbing is connected appropriately at all times, and to suppress all manual prompts to set valves.

The **CGR.ini** file's content remembers just the Size and Location and DatPath of the "central control" form of ***Group Run***. The keyed parameter (ComDisp=0) is a legacy parameter meaning that the form has associated with it *separate module windows* for each module being processed (old value = 1, meaning *single common form* is no longer used, and must not be set).

```
--- NUSS 'Coord.Group Run Form' Initialization File:  (All Modules)
--- Using:     NUSS Version: 0.1.5
--- Updated: 09-05-2000 at 12:37:44
Size= 2580 5790
Location= -30 975
DatPath= (D:\Nuss\Src\Dat\)
ComDisp= 0
```

This file has no module-specific information, but each modules <*moduleid*>**R.ini** file is used instead.

F.2.3 Run State Definition Files for Modules

There is at least one Run State Definition (*<modid>RS<runstate#*>**.ini)** file for each single module – as each module has at least a default Run State 0. These files are located in the **Ini** subfolder of the NUSS install path (see item 7 in the NUSS File Summary Table). Each file defines a particular Run State (0-9) of a module, and is created and edited with the **Run State Editor** (except the *non-editable default* **Run State 0**, created by the ***Run*** function the first time a new module runs).

```
--- Run State 0's Initialization for Module: 9021-141 () Scanner
--- Module Firmware Ver:   2.05
--- Using:     NUSS Version: 0.1.1
--- Updated: 08-08-2000 at 14:07:01
UserName= (Default)
AcqState=  1
RecState= -1
CmdP00= © 00 1 0fff 1 0 7 1)
CmdP00= © 00 2 0fff 1 0 7 1)
CmdP00= © 00 3 0fff 1 0 7 1)
CmdG05= © 05 1 0010)
CmdG05= © 05 2 0060)
CmdG05= © 05 3 0380)
CmdS00= © 00 1 0fff 1 500 7 0)
CmdS00= © 00 2 0fff 1 2000 7 0)
CmdS00= © 00 3 0fff 1 7500 7 0)
EndFunc= -1
Average= 8
IniDSet= 0
DispSet= 0 &HFFF All Chan
```

The first item (UserName) assigns a *name* string to the **Run State**. Keys AcqState and RecState specify the *Start* conditions of the Run State (-1=Null (don't care), 0=Off, 1=On) for ***Acquire*** and ***Record*** features of ***Run*** or ***Group Run*** windows, respectively.

Two or three groups of three commands each follow (with keys CmdXYY, where YY selects the **00** or **05** stream initialization sub-commands of command "**c**", and X describes the group). The first group (P) initializes Prefix streams if any (it is missing otherwise). The next group selects Data Group (G) content for each stream. The final group (S) initializes normal Sustaining streams. Due to embedded blanks, parentheses must surround each command.

The key EndFunc defines a **Run State** to *chain-to* when this stream ends (-1 = None). The key Average defines the "# A/D Scans to Average" option to be sent to module when this Run State starts.

The keys IniDSet and DisplSet currently specify that only **Display Set 0** is to be used with this **Run State** (all channels scanned are displayed, in order of channel #). In the future, they could allow other **Display Sets** to exist for each **Run State** (i.e., other subsets of the scanned channels may be displayed, in any order, on demand).

F.2.4 Display Set Definition Files for Modules

There is at least one **Display Set** definition (<*moduleid*>**RS<***runstate#***>.0)** file for each single module – as each module has at least a default **Run State 0**, and each **Run State** has exactly one **Display Set** (designated 0). The original design of NUSS allowed for additional **Display Sets** per **Run State** but this feature was never implemented. Thus, only this single *default* **Display Set 0** exists for each **Run State**. These files are located in the **Ini** subfolder of the NUSS install path (see item 8 in the NUSS File Summary Table). Each file defines the *default* **Display Set** used whenever a particular **Run State** (0-9) is selected for a module.

```
--- Display Set 0's Initialization for Module: 9021-141 () Scanner
--- Module Firmware Ver:   2.05
--- Using:     NUSS Version: 0.1.1
--- Updated: 08-08-2000 at 15:25:31
---  Ch/Rng  EU: Nam/Typ/Prc/Un/AScal/AUn  Oth: Nam/Typ/Prc/Un/AScal/AUn (RngDesc)
InfoCh1= 1 29   P1 0 3 psia 1 ()    T1 2 2 degC 1 ()      (29= 2.5 to 100 psia)
InfoCh2= 2 29   P2 0 3 psia 1 ()    T2 2 2 degC 1 ()      (29= 2.5 to 100 psia)
InfoCh3= 3 0    X3 0 5 V    1 ()    Y3 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
InfoCh4= 4 0    X4 0 5 V    1 ()    Y4 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
InfoCh5= 5 0    X5 0 5 V    1 ()    Y5 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
InfoCh6= 6 0    X6 0 5 V    1 ()    Y6 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
InfoCh7= 7 0    X7 0 5 V    1 ()    Y7 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
InfoCh8= 8 0    X8 0 5 V    1 ()    Y8 2 2 V    1 ()      (0= -5 to 5 V or 0 to 20 mA)
GraphOn= 0 255 40 12 30 6316128
```

This file determines all *display parameters* of the module's ***Run*** or ***Group Run*** "display" form when using the associated **Run State**. This includes the content and format of each *tabular* display field on each *data display line*, for both its *EU datum* and its *"other" datum*, and the *range* of the transducer assigned to that data line. **Display Set 0** has such a line for *each channel currently scanned* by the module's **Run State** (worst case of all three streams). In this example, only 8 of the Model 9021's 12 channels are included. Presumably, channels 9-12 were not specified in any stream. Each InfoChX keyed data line (where X refers to a particular ***Run*** display data line) contains basically *four columns* of parameters. The *first column* contains 2 parameters: the *transducer #* assigned to that data line and its *range code*. The *second column* contains 6 parameters for the *EU tabular datum*: its *name*, *data type*, *precision*, *units string*, *alternate scale factor*, and *alternate units string* (shown as *empty string* () when NO alternate units are assigned). The *third column* replicates the previous one exactly, but applies to the "other" datum for that data line. The *fourth column* contains a text string description of the transducer range code's *low* and *high* range values (must be within parentheses because of embedded blanks). It is used to select a suitable graphic range for bar and trend graphs for this transducer. Also notice, for this Model 9021 example, that only the first two data lines have a range code. This indicates that 9400 pressure transducers were plugged into these data line's channels. No transducers were apparently plugged in to the other channels, causing them to display *voltage* for both EU and "other" datum fields.

The final key GraphOn contains the *graph state* (Bar, Trend, None) and all other graphic parameters (currently for the Trend Graph).

F.2.5 Group Definition Files

There is only one type of Group Definition (*G<grpid*>**.ini)** file. Such files are creations of the **Group Editor**, and are located in the **Ini** subfolder of the NUSS install path (see item 9 in the NUSS File Summary Table).

```
--- NUSS 'Group A' Initialization File:  (All Modules)
--- Using:     NUSS Version: 0.1.1
--- Updated: 08-16-2000 at 10:41:10
UserName= (Byron's Grp)
ModID= 9034 119
ModID= 9816 284
ModID= 9016 669
```

The UserName key defines any verbose 16-byte name given to the Group. This string is always in parentheses due to the possibility of embedded blanks (it must be treated as a single string parameter).

Zero or more lines follow with ModID keys, each defining the <*model#*> and <*serial#*> of a particular module assigned to this group by the editor.

The special **Group *** (Star), which represents a *dynamically-changing group* called “All Connected Modules”, is not defined in a Group Definition File. Thus, there is no **GStar.ini** file in the **Ini** subfolder. However, there may can be *recorded data files* (and corresponding *run-number initialization* files) located in the **Dat** subfolder, that do use the “GStar” name.

F.2.6 Logical Range Association File

There is only one Logical Range Association (**LRN.ini)** file. It is located in the **Ini** subfolder of the NUSS install path (see item 10 in the NUSS File Summary Table).

```
--- NUSS 'LRN' Initialization File:  (All Modules)
--- Using:     NUSS Version: 2.0.6
--- Updated: 07-03-2007 at 14:33:27
---      = LRN Mod-ID (Range) Delay ZeroPt SpanPt #MultiPts Pt1 Pt2 ... CalAbs Sclr
LRNAssign= 1 9034-0 (-83 to 104 kPa(*)) 6 0 104 5 -83 0 104 0 -83 False 6.894757
LRNAssign= 2 9038-0 (-5 to 5 psi) 6 0 5 9 -5 -2.5 0 2.5 5 2.5 0 -2.5 -5 False 1
LRNAssign= 3 9034-0 (0 to 207 kPaA) 6 0 104 3 -83 0 104 True 6.894757
LRNAssign= 4 Extern. (-200 to 1300.0 degC) 6 -99999 1300 0 False 0
LRNAssign= 5 Extern. (0 to 5000 ohm) 6 0 0 3 0 0 0 False 0
LRNAssign= 6 Extern. (-200 to 1300.0 degC) 6 0 0 0 False 0
RepOpt= 217 () () () () ()
```

This file is written when the ***'Configure | Calibrators (NUSS)'*** form is exited normally (i.e., [OK] clicked instead of [Cancel] key).

This file contains zero or more (up to 8) data lines with the LRNAssign= key. These each assign a unique LRN (Logical Range Number), 1-8, that refers to some Calibrator module that can suitably provide pressure (or other) data set points to calibrate various associated channels of one or more scanner module with a particular range. Preceding these keys is a legend line that labels each parameter following the equal sign (=) on the corresponding keyed line, the *first parameter* being the *LRN* (logical range number) itself.

The *second parameter* is the *Mod-ID* (i.e., the Calibrator module that sets all the LRN's set points) identified by a <*moduleid*> parameter, which is always <model#>-<serial#> for NetScanner Calibrators, but would be <8400>-<icrs> for System 8400 PCU Calibrators (Advanced feature). See end of NUSS File Summary Table for *icrs* parameter format.

The *third parameter* is a 4-token *Range* string (always in parentheses, because it has embedded spaces). Tokens 1 and 3 show the *low* and *high* range end points of that LRN (possibly edited). Token 3 is just a separator ("to") but is required. Token 4 is the name of the units of measure of each set point.

The remaining LRN parameters (4th, 5th, ... last) include:

- *Delay* or Settle Delay Time in seconds after each set point is set by calibrator, but before data is read from the calibrator or the module channels being calibrated),
- *ZeroPt*, or the Zero-Only set point,
- *SpanPt*, or the Span-Only set point,
- *#MultiPts*, or Number of Multi-Point set points to follow
- *Pt1, Pt2, ...* or the values of each of those set points in the units specified by the 4th *Range* token.

- *CalAbs*, or the Absolute Mode parameter, a boolean that is True if all the preceding set points are in *Absolute* pressure units, is False for Differential pressure mode (also used for set points in any other *non-pressure* units).
- *Sclr* (the final parameter), or the E.U. Conversion Factor (or Scalar) of the calibrator. For pressure calibrators only (real or External) it is **=1** for calibrators that set pressure in psi units, or **=another non-zero value** for other pressure units. It is a multiplier used for converting any other std. pressure units into psi. This multiplier for an External calibrator with non-pressure units, like temperature (degC or degF units) or resistance calibrator (ohm units), has a value of zero (**=0**).

The file ends with one line with the RepOpt= key. These save the various Common Options and other test options (see bottom of ***'Configure | Calibrators (NUSS)'*** form).

F.2.7 Logical Barometer Association File

There is only one Logical Barometer Association (**LBN.ini**) file. It is located in the **Ini** subfolder of the NUSS install path (see item 11 in the NUSS File Summary Table).

```
--- NUSS 'LBN' Initialization File:  (All Modules)
--- Using:     NUSS Version: 0.1.1
--- Updated: 08-14-2000 at 15:41:51
LBNAssign= 1 9032-0 (0 to 30)
LBNAssign= 2 9034-0 (0 to 5)
```

This file contains zero or more (up to 4) data lines with the LBNAssign key. These each assign a unique LBN (Logical Barometer Number), 1-4, to some Barometer module (an *absolute standard* or an *absolute calibrator*).

The module is identified by a <*moduleid*> parameter, which is always <model#>-<serial#> for NetScanner Standard or Calibrator modules, but would be <8400>-<*icrs*> for System 8400 PSU Standards or PCU Calibrators (Advanced feature). See end of NUSS File Summary Table for *icrs* parameter format.

The last parameter is a single string (in parentheses) showing the *low* and *high* range of that Standard's or Calibrator's natural range.

F.2.8 Overall NUSS Memory File(s)

There is normally only one Overall NUSS Memory (**NUSS.ini)** file by default, but multiple copies with names [xxx]**NUSS.ini** may exist. The file names vary in the leading characters ([xxx]). These files are located in the **Ini** subfolder of the NUSS install path (see item 12 in the NUSS File Summary Table and See **Section 3.5** in Chapter 3 for more information).

```
--- NUSS 'Network Status' Initialization File:  (All Modules)
--- Using:     NUSS Version: 1.1.19
--- Updated: 01-13-2003 at 08:35:54
MainSize= 1095 5685
MainLocation= 420 90
NetSize= 6540 8265
NetLocation= 1515 90
InsDirName= (C:\Nuss\Src\)
Group=  0
ViewPath= (C:\Program Files\JGsoft\EditPadPro\EditPadPro.exe)
ManPath= (S:\RELEASED\9000\Nuss\UserMan\nussfac.htm)
ArcPath= (u:\Archive\)
EthSysIP= False ()
EthSysPCUcrs= ()
EthSysReset= False
GpibSysDev= False (PSI)
GpibSysPCUcrs= (1800@113 835@115 448@117)
GpibSysReset= True
GpibKdvDev= False ()
GpibPvrDev= False ()
GpibTcoDev= False ()
DiagFunc= 0
OthFacOpt= 0
```

This first two keyed data lines remember the Size and Location of the *main* NUSS (i.e., **home-base**) form when NUSS last exited. Likewise, the next two keyed data lines remember the Size and Location of the **Network Status** form.

The key InsDirName= remembers the Install file path of NUSS. It is this base path from which other standard NUSS subfolders (**Ini**, **Dat**, **Report**, **Firmware**) are defined. This is a single string parameter (possibly containing embedded spaces) thus must be in parentheses.

The Group= key remembers the *Current Group* assigned when NUSS last exited.

The ViewPath=, ManPath=, and ArcPath= keys remember the configured paths of the NUSS Text Editor, where the NUSS user's manual is installed, and the Archive Base Path, which are all parameters of the ***'Configure | General Options (NUSS)'*** home-base menu item.

All remaining keys in the file are used to remember *Advanced* and *Factory* user options on the form displayed for this same menu item (i.e., special Ethernet or GPIB devices configured, as well as other special diagnostic options). For *Normal* users, these extra items are missing from the file.

F.2.9 Playback Secondary Output Files and Secondary ID Files

Recorded data (**.Dat**) files (see NUSS File Summary Table items 1 and 2) are generated by the ***Record*** feature of both the *single-module* ***Run*** and ***Group Run*** functions. Each of these files may be “played-back” *numerous times* by the complementary ***Playback*** feature. During *each* such playback operation, a *unique* **Playback Secondary Output File** may be created (see NUSS File Summary Table item 13) that contains either *all* or *edited parts* of the original binary data file, but reformatted into a readable text report. These *possibly many* report files are *automatically named* to avoid confusion. This is done by appending a variable length **Secondary ID** string (<*sid*> = 1 to n upper-case alphabetic characters) to the file name, which otherwise has the exact same *root name* as the *recorded data file* from which it was generated. This <*sid*> string appendage, which comes from another **.ini** file (see NUSS File Summary Table item 14), is “incremented alphabetically” for each playback operation of the recorded data file that creates any secondary reports. In other words, if the *source* recorded data file being played has the name:

9016-333-123.dat (From Model 9016 module, serial #333, test run #123)

then the first (and subsequent) **Playback Secondary Output** file(s) generated during multiple play-backs would be named as follows:

9016-333-123A.txt (1st report file from playing 9016-333-123.dat)
9016-333-123Z.txt (26th report file from playing 9016-333-123.dat)
9016-333-123AA.txt (27th report file ...)
9016-333-123AZ.txt (52th report file ...)
9016-333-123BA.txt (53th report file ...)
: **:** **:**
9016-333-123ZZZZZ..Z.txt (ultimate last report file with longest name...)

As previously indicated, the format of these formatted report files is pure readable text, and such files can be examined (on screen) or printed by any text processing application that can handle the (possibly long) text lines in them (i.e., for specific autonomous stream dumps with large numbers of channels in a group). See **Chapter 6** for information on interpreting these files.

The Secondary ID **.ini** file used in automatically naming these report files is quite simple. If no such file exists (with the root name of the played back recorded data file concatenated to the string “**SID.ini**”) then it is initially created with only a single line of text containing:

“A”

After this string is used to name the first report, it is “incremented” to “B” and rewritten to the **.ini** file. When the next report file from same source file is opened, the “B” is used to create that unique report name, and it is again “incremented” and rewritten to the **.ini** file. The first alphabetic sequence (A..Z) ends after the 26th report and wraps “Z” back to “A”, after which a new character is added (e.g, “AA”) to the **.ini** file’s single text line. Each “alphadigit” increments in positional notation until it becomes “ZZ”, after which “AAA” is created, and so on. There is no limit to this algorithm except for the ultimate file name length limits of Win32.

> **Important Note:** When performing any necessary file maintenance on **Playback Secondary Output** files in the **Dat** subfolder (of NUSS install path), it is important to remember that there is a <*playbackfilename*>**SID.ini** file for each <*playbackfilename*>**.dat** file ever played back. If you playback a different recorded data file, a whole new report file naming sequence (starting at “A”) begins for it uniquely, using a unique **.ini** file. After groups of these reports are removed (e.g., deleted or moved) from this subfolder, it is important that you **delete** the associated data file’s appropriate **.ini** file so that automatic naming continues — per a new sequence of such files.

F.2.10 Test/Calibrate Program Tolerance File

This tolerance file is shared by the special ***Leak & Pressure Set Accuracy Test*** program for 903x calibrators modules, and by the ***Leak Test***, ***Pressure Accuracy Test***, and ***Calibration Adjustment*** forms for scanner modules. The file is named **TAccCal.ini** and is located in the **Ini** subfolder. A factory supervisor may modify this file if necessary to alter the tolerances of these tests. A copy of the default factory file is listed below. As can be seen from the comment (third field each line) some tolerances apply to only specific module types or transducer types

```
--- NUSS 'Calibration & Press.Acc.Test' TOLERANCES File
SETDELTA=      0.9       Setpoint Delta (deg.C) for Oven Temp.to be declared Set
SETSPEC=       0.01      Setting Pressure Accuracy Tolerance (%FS Calibr.Hi Range)
CLPSPEC=       0.2       Clamping Pressure Accuracy Tolerance (%FS Calibr.Hi Range)
ACCSPEC1=      0.15      9x16 Press.Acc.Tolerance (%F.S.Reading) for Std.Ranges <= 2.5 Psi
ACCSPEC2=      0.05      9x16 Press.Acc.Tolerance (%F.S.Reading) for Std.Ranges > 2.5 Psi
ALTACCSPEC1=   0.05      9x16 Press.Acc.Tolerance (%F.S.Reading) for MDS Ranges <= 5 psi
ALTACCSPEC2=   0.025     9x16 Press.Acc.Tolerance (%F.S.Reading) for MDS Ranges > 5 psi
READSPEC=      0.25      9x16 Press.Acc.Tolerance (% of Reading) for All MDS Ranges
ACCSPEC3=      0.2       9021 Press.Acc.Tolerance (%F.S.Reading) for Std.Ranges <= 2.5 psi
ACCSPEC4=      0.1       9021 Press.Acc.Tolerance (%F.S.Reading) for Std.Ranges > 2.5 Psi
CUMLOERROR=    0.25      Cumulative Zero Tolerance (%Calc.Value) for rgs <= 2.5 psi
CUMHIERROR=    0.05      Cumulative Zero Tolerance (%Calc.Value) for rgs > 2.5 psi
LEAKSPEC=      2         Leak Tolerance (%FS) for full LEAKTIME sec. Check Duration
LEAKTIME=      120       Standard Leak Test Duration (seconds)
LEAKSPECPURGE=               5            Purge Leak Tolerance (%FS of LRN)
LEAKTIMEPURGE=               180          Purge Leak Test Duration (seconds)
CALSETTIMEOUT=               60           Calibrator Setting Timeout (seconds)
```

When used by the two "leak check" programs, the LEAKSPEC (2 %FS) is adjusted up or down dynamically – if the LEAKTIME is changed from its default (120 second) value. The LEAKSPECPURGE and LEAKTIMEPURGE are used only for special Purge Leak test. CALSETTIMEOUT is used currently only by the Voltage Accuracy Test program.

F.2.11 Compensation Accuracy Test Tolerance File

This is a legacy tolerance file used only by the ***Compensation Accuracy Test*** program. It is a simple text file in the **Ini** subfolder, with the name **Tcomp.ini**. A factory supervisor may modify this file if necessary to alter the tolerances of this test. A copy of the default factory file is listed below. It begins with two text lines with a single number on each line, followed by a "comment" text line that describes each number line.

```
 0.25
 0.25
order should be zero coef.tolerance, then span tol.in terms of % of FS
```

F.2.12 Range Codes For Transducers File

The **RngCodes.ini** file is written to the **Ini** subfolder whenever NUSS starts **– if it does not already exist**. Thereafter, a *supervisor* user may edit it to add special User Range codes (for keys 100= through 144=) for special uses – or to correct Factory Range codes (for keys 0= through 64=) that you are advised need updating. The middle codes (65= through 99=) are used only by the **[Help]** feature on the ***'Configure | Calibrators'*** form, for setting appropriate special customer-specific calibration set points used by some test programs.

This *range end-point* and *units* information, looked-up in this file using each transducer's assigned (at factory) Range Code, is used by NUSS in various ways for all pressure scanner modules. This information is also used to set the High scaling end-point of that transducer's bar graph on ***Run*** forms, when a particular channel's DH or Model 9400 transducer is found to have one of the known Factory Range codes. The Low scaling end-point of a bi-polar bar graph is always set to the negative of the High scaling end-point value (to insure that the zero point is always in the center of the bar graph), even though the actual transducer Low end-point (obtained from this file) may be a lessor negative number. A copy of the default factory version of the file is listed below.

```
-1= 903x Has No Range Code
0= -5 to 5 V or 0 to 20 mA
1= -0.36 to 0.36 psi or -10 to +10 inH2O
2= -0.72 to 0.72 psi or -20 to +20 inH2O
3= -1 to 1 psi
4= -2.5 to 2.5 psi
5= -5 to 5 psi
6= -5 to 10 psi
7= -5 to 15 psi
8= -5 to 30 psi
9= 0 to 45 psi
10= 0 to 100 psi
11= 0 to 250 psi
12= 0 to 500 psi
13= 0 to 600 psi
14= 0 to 300 psi
15= 0 to 750 psi
16= -10 to 10 psi
17= -12 to 15 psi
18= -12 to 30 psi
19= -12 to 45 psi
20= -12 to 20 psi
21= 0 to 20 psi
22= 0 to 15 psi
23= -10 to 15 psi
24= 0 to 5 psi
25= 0 to 10 psi
26= 0 to 30 psi
27= 0 to 50 psi
28= 0 to 100 psi
29= 2.5 to 100 psia
30= 25 to 250 psia
```

```
31= 2.5 to 50 psia
32= 25 to 500 psia
33= 25 to 750 psia
34= 2.5 to 30 psia
35= 2.5 to 15 psia
36= 0 to 125 psi
37= -12 to 35 psi
38= 0 to 150 psi
39= 0 to 200 psi
40= -12 to 22 psi
41= -12 to 60 psi
42= 0 to 375 psi
43= 0 to 150 psi
44= 0 to 75 psi
45= 0 to 150 psi
46= 0 to 650 psi
47= 0 to 850 psi
48= 25 to 150 psia
49= 50 to 750 psia
50= 2.5 to 75 psia
51= -1.2 to 1.2 psi
52= 0 to 1000 psi
53= 0 to 1000 psia
54= 0 to 1500 psi
55= 0 to 1500 psia
56= 0 to 3000 psi
57= 0 to 3000 psia
58= 0 to 5000 psi
59= 0 to 5000 psia
60= 0 to 10000 psi
61= 0 to 10000 psia
Customer MDS RAMS
65= -7 to 0 psi
66= -10 to 0 psi
67= -0.36 to 0 psi
68= -5 to 0 psi
69= -1 to 0 psi
70= 0 to 120 psi
71= 0 to 320 psi
72= 0 to 650 psi
73= 0 to 15 psi
Customer Other (100-144)
End
```

NOTICE: After changing this file, the NUSS application must be exited and restarted before the new file is recognized by it. Also, any module that has a Range Code changed in this file needs to have the FORGET *context menu* item executed for it before any new range/units information is recognized by that module for the affected transducers.

F.2.13 Sensor Types for Model 9x46 Modules File

The **SensorType9046.ini** file is written to the **Ini** subfolder whenever NUSS starts **– if it does not already exist**. It is similar to the Range Codes file for pressure scanners, but is specifically for the Model 9x46 temperature scanners only. A *supervisor* user may edit it to add special Sensor Type codes to correct Factory codes that you are advised need updating – or just to change the arbitrary ranges of the defaults.. Normally, the range and units information is used by NUSS in various ways for all pressure scanner modules, but principally to choose a correct scaling of bar graphs on **Run** forms, when a particular 9x46 channel's configured temperature or resistance transducer is found to have one of the known sensor type codes. A copy of the default factory version is listed below.

```
-1= 9046 Has No Sensor Code
0= -5 to 5 V V
1= -5 to 5 V P
11= 0 to 600 degC B-TC
12= -270 to 300 degC E-TC
13= -210 to 300 degC J-TC
14= -270 to 300 degC K-TC
15= -270 to 300 degC N-TC
16= -50 to 300 degC R-TC
17= -50 to 300 degC S-TC
18= -270 to 300 degC T-TC
30= 0 to 250 Ohm R250
31= 0 to 500 Ohm R500
32= 0 to 1000 Ohm R1K
33= 0 to 20000 Ohm R20K
40= -80 to 200 degC Th3K
41= -80 to 200 degC Th5K
42= -80 to 200 degC Th10K
60= -200 to 850 degC RTD385
65= -200 to 850 degC RTD7990
Customer Other (75-100)
End
```

The fourth parameter (after units) on each line, is not used by any NUSS function, but serves as a comment parameter to label its particular Sensor Type. Here, **<type>-TC** means Thermocouple of *type* B, E, J, K, N, R, S, T. **Th<resrange>** means Thermistor, **RTD<type>** means Resistance Temperature Device, and **R<resrange>** means Resistance.

NOTICE: After changing this file, the NUSS application must be exited and restarted before the new file is recognized by it. Also, any module that has a Sensor Code changed in this file needs to have the FORGET *context menu* item executed for it before any new range/units information is recognized by that module for the affected transducers.

F.3 All Report Files And The Forms Creating Them

Report files are all written to the NUSS installation subfolder **Report**. Such files all have the **.txt** extension, though some may be duplicated with the **.csv** extension. The names of all human-readable text report files generated by the *Configure, Calibrate*, and *Test* forms are listed in this section. Examples of most follow in sections **F.4** and **F.5**. The menu function (form) that generates each file is also shown.

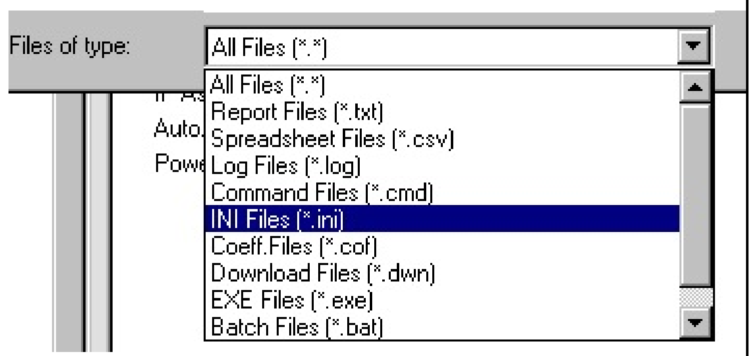
Menu Function	Report File Name
Calibrate | Multipoint Calibrate | Zero-Only Calibrate | Span-Only	***CAdj.txt** (or **.csv**) ***CAdjZ.txt** (or **.csv**) ***CAdjS.txt** (or **.csv**)
Calibrate | View/Edit Coefficients	***CCoef.txt**
Test | Leak (6 modes): Auto-Full-Ckt-LRN mode: Cal mode: CalRef mode: Run mode: RunRef mode: Run-Leak-Option mode:	***TLk**<*modename*>**.txt** (or **.csv**) where: <*modename*> = "Auto" (**TLkAuto**) <*modename*> = "C" (**TLkC**) <*modename*> = "CR" (**TLkCR**) <*modename*> = "R" (**TLkR**) <*modename*> = "RR" (**TLkRR**) <*modename*> = "RLO" (**TLkRLO**)
Test | Pressure Accuracy (scanners)	***TAcc.txt** (or **.csv**)
Test | Pressure Set Accuracy (calibrators)	***TPrSet.txt** (or **.csv**)
Test | Compensation Accuracy (scanners)	***TComp.txt** (or **.csv**)
Test | Noise	***TNoise.txt** (or **.csv**)
Test | Memory (scanners with EEPROM)	***TMemory.txt** (or **.csv**)
Configure | Network Options	***CfgNet.txt**
Configure | Transducer Options	***CfgXdcr.txt**
Configure | Other Options	***CfgOth.txt**

The ***** (in the **Report File Name** column above) is a *wild-card* that stands for the string "***<modid>***" for a *Single-Module* report, or the string "***CG<grpid>***" for a *Coordinated Group* report containing the data from several modules.

Other tests (not listed above and in Chapter 5) may appear in the separate Chapter 5 and Appendix F of the *Advanced* addendum

Please note that many of the report files above may also be optionally duplicated in a special *spreadsheet-compatible* format called "comma-separated-variable" files. Such files have the **.csv** file extension.

Also in this **Report** subfolder you may find a few **.ini** files created only by the **Pressure Accuracy Test**. Such files are named **<*modid*>TAcc.ini** where <*modid*> is any identifier of a particular pressure scanner module (e.g., 9116-1234).



To view the contents of any of these **.ini** files use the **.ini** file *filter* on the *dialog* form that pops-up when you select the menu item ***"File |View Files (NUSS)"*** from the *home base* form. Be sure you select the **Report** subfolder instead of the **Ini** subfolder in this case.

These files are simple text files with four header lines followed by a series of keyed data lines, one for each channel of the module (see example below for simulated pressure scanner module 9016-0).

```
--- NUSS 'Cumulative Zero' Initialization File for Module: 9016-0 () Scanner
--- Module Firmware Ver:   2.42
--- Using:     NUSS Version: 2.0.10
--- Updated: 10-22-2007 at 14:33:15
Ch= 1 0.0001 2
Ch= 2 0.0002 2
Ch= 3 0.0003 2
Ch= 4 0.0004 2
Ch= 5 0.0005 2
Ch= 6 0.0006 2
Ch= 7 0.0007 2
Ch= 8 0.0008 2
Ch= 9 0.0009 2
Ch= 10 9.999999E-04 2
Ch= 11 0.0011 2
Ch= 12 0.0012 2
Ch= 13 1000000 0
Ch= 14 1000000 0
Ch= 15 1000000 0
Ch= 16 1000000 0
```

Each data line has the keyword "**Ch=**" followed by *three (3)* space-delimited data values (left to right) :

- the *channel number*,
- the *base zero/offset coefficient* (recorded after first ReZero session), and
- the *number of ReZero operations* that have occurred over all sessions.

Note that in this example the first 12 channels of the module have had a successful *Re-Zero* operation performed twice (third parameter = 2), while the remaining channels (13-16) have never had such a successful test performed (third parameter = 0). The value 1000000 is a placeholder for the missing coefficients which indicates the "cleared" state, when no previous zero/offset coefficient value exists. These remaining "cleared" channels may simply be associated with some different LRN which is assigned to a different pressure range that has not yet been tested. If these channels are tested later the file may then contain real zero/offset coefficients with history counts greater than zero (after the test program exits only).

F.4 Full *Calibration Adjust* Report File Examples from Chapter 4

Several example *calibration adjustment* reports (that were once included in Section 4.6 of Chapter 4) have been moved here.

In this first example report for a *Multi-Point* adjustment session, there is data only for a single LRN with five (5) set points retained in the report. Data for a second LRN would appear sequentially in the next section of the report, as indicated. This example shows a report when the native units of the calibrator and the calibrated scanner channels were all set to psi native units. However, the report would be the same for any other native units (like kPa) except for the unit names that appear in the report.

```
--- NUSS 'Calibration Adjustment Report' for a Single Module:
----  9016-0 () Scanner associated with LRN 1,2 (Scans/Avg=8 Sfw.Ver.=2.24)
--- Session is adjusting calibration Multi-Points
--- Report generated: 10-25-2007 at 15:14:06 using NUSS Version: 2.0.11
--- Contents include Summary Results only

  Adjusting Offset+Gain For 1 Scanner Module using LRN 1 (9034-0) :

    Set Calibrator to DIFFERENTIAL Mode
        Scanners' Offset+Gain coefficients were adjusted.

      Indicated Readings: LRN 1 @ Pt. 1          2            3            4            5
        Calibr. 9034-0      Set to:  -34.0        -17.0        0.0          17.0         34.0
          :       :         Reads:   -34.0        -17.0        0.0          17.0         34.0
        Scanner 9016-0      Ch        Pr-kPa(*)...
          :       :         1        -33.999      -16.999      0.001        17.001       34.001
          :       :         2        -33.998      -16.998      0.002        17.002       34.002
          :       :         3        -33.997      -16.997      0.003        17.003       34.003
          :       :         4        -33.996      -16.996      0.004        17.004       34.004
          :       :         5        -33.995      -16.995      0.005        17.005       34.005
          :       :         6        -33.994      -16.994      0.006        17.006       34.006
          :       :         7        -33.993      -16.993      0.007        17.007       34.007
          :       :         8        -33.992      -16.992      0.008        17.008       34.008
          :       :         9        -33.991      -16.991      0.009        17.009       34.009
          :       :         10       -33.99       -16.99       0.01         17.01        34.01
          :       :         11       -33.989      -16.989      0.011        17.011       34.011
          :       :         12       -33.988      -16.988      0.012        17.012       34.012

      Adjustment Summary: LRN 1   Offset       Offset       Gain         Gain         Range
        Scanner 9016-0      Ch    (orig.)      (final)      (orig.)      (final)      (F.S.)
          :       :         1     0.0          0.0          1.0          1.0          34 kPa
          :       :         2     0.0          0.0          1.0          1.0          34 kPa
          :       :         3     0.0          0.0          1.0          1.0          34 kPa
          :       :         4     0.0          0.0          1.0          1.0          34 kPa
          :       :         5     0.0          0.0          1.0          1.0          34 kPa
          :       :         6     0.0          0.0          1.0          1.0          34 kPa
          :       :         7     0.0          0.0          1.0          1.0          34 kPa
          :       :         8     0.0          0.0          1.0          1.0          34 kPa
          :       :         9     0.0          0.0          1.0          1.0          34 kPa
          :       :         10    0.0          0.0          1.0          1.0          34 kPa
          :       :         11    0.0          0.0          1.0          1.0          34 kPa
          :       :         12    0.0          0.0          1.0          1.0          34 kPa

  Adjusting Offset+Gain For 1 Scanner Module using LRN 2 (9038-0) :

    Set Calibrator to DIFFERENTIAL Mode
        Scanners' Offset+Gain coefficients were adjusted.

      Indicated Readings: LRN 2 @ Pt. 1          2            3            4            5
        Calibr. 9038-0      Set to:  -34.0        -17.0        0.0          17.0         34.0
          :       :         Reads:   -34.0        -17.0        0.0          17.0         34.0
        Scanner 9016-0      Ch        Pr-kPa...
          :       :         13       -33.999      -16.999      0.001        17.001       34.001
          :       :         14       -33.998      -16.998      0.002        17.002       34.002
          :       :         15       -33.997      -16.997      0.003        17.003       34.003
          :       :         16       -33.996      -16.996      0.004        17.004       34.004
```

```
      Adjustment Summary: LRN 2   Offset       Offset       Gain         Gain         Range
        Scanner 9016-0      Ch    (orig.)      (final)      (orig.)      (final)       (F.S.)
         :       :          13    0.0          0.0          1.0          1.0         *6.89 kPa
         :       :          14    0.0          0.0          1.0          1.0         *6.89 kPa
         :       :          15    0.0          0.0          1.0          1.0         *2.48 kPa
         :       :          16    0.0          0.0          1.0          1.0         *2.48 kPa
         NOTE: Low Press.Ranges (<=1 psi) are marked * above.

Adjustment(s) Completed OK.
```

An example of a Zero-Only calibration adjustment report for a Model 9x46 temperature or resistance scanner is shown below. In this case three different LRN's are assigned to various channels of the module. Each LRN calibrates a particular sensor type (or group of functionally-similar sensor types). The first (LRN #4) is assigned to a several thermocouple (TC) channels of different types – and uses the **Auto-UTR** offset adjustment method. The second (LRN #6) is assigned to three *resistance* channels – and uses a *simple* offset adjustment method. The third (LRN #7) is assigned to eight thermocouples – and uses the **Thermal Bath** offset adjustment method (NOTE: this group could have also been mixed with sensors of other thermal types like RTD's or Varistors). The rather "mixed results" obtained in this 3-part report were the result of operator errors and other problems. This was done on purpose to illustrate problems an operator might experience in practice. How to avoid these pitfalls is discussed after this listing of the 3-part report.

```
--- NUSS 'Calibration Adjustment Report' for a Single Module:
----  9046-395 () Scanner associated with LRN 4,7,6 (Scans/Avg=64 Sfw.Ver.=2.43)
--- Session is adjusting calibration Zero-Only
--- Report generated: 06-29-2007 at 09:46:40
--- Contents include Summary Results only

  Adjusting Offset For 1 Scanner Module using LRN 4 (Extern.) :

             LRN #4 is assigned to an EXTERNAL Temperature Calibrator.
              You are PROMPTED to perform special tasks MANUALLY that
          are important to obtaining a successful calibration adjustment!

        Scanners' Offset coefficients were adjusted.

      Indicated Readings: LRN 4 @ Pt.Zero-Only
        Calibr. Extern.     Set to:  Auto-UTR per-ch. values
         :       :          Reads:   Auto-UTR per-ch. values
        Scanner 9046-395    Ch        T-degC...
         :       :          1         22.84587
         :       :          2         25.06712
         :       :          3         27.46433
         :       :          5         25.47197
         :       :          6        ********

      Adjustment Summary: LRN 4   Offset       Offset      Range
        Scanner 9046-395    Ch    (orig.)      (final)     (F.S.)
         :       :          1     0.0         -4.530159    300 degC
         :       :          2     0.0         -2.391337    300 degC
         :       :          3     0.0         -0.043659    1350 degC
         :       :          5     0.0         -2.181618    1350 degC
         :       :          6     0.0          99971.29    1350 degC
```

```
  Adjusting Offset For 1 Scanner Module using LRN 7 (Extern.) :

            LRN #7 is assigned to an EXTERNAL Resistance Calibrator.
            You are PROMPTED to perform special tasks MANUALLY that
         are important to obtaining a successful calibration adjustment!

      Scanners' Offset coefficients were adjusted.

    Indicated Readings: LRN 7 @ Pt.Zero-Only
      Calibr. Extern.     Set to:   0.0
       :       :          Reads:    0.0
      Scanner 9046-395    Ch        R-ohms...
       :       :          4         220.772
       :       :          7        ********
       :       :          8         109.0231

    Adjustment Summary: LRN 7   Offset       Offset      Range
      Scanner 9046-395    Ch    (orig.)      (final)     (F.S.)
       :       :          4     0.0          220.772     500 Ohm
       :       :          7     0.0          10000000.0  500 Ohm
       :       :          8     0.0          109.0231    500 Ohm

  Adjusting Offset For 1 Scanner Module using LRN 6 (Extern.) :

           LRN #6 is assigned to an EXTERNAL Temperature Calibrator.
            You are PROMPTED to perform special tasks MANUALLY that
         are important to obtaining a successful calibration adjustment!

      Scanners' Offset coefficients were adjusted.

    Indicated Readings: LRN 6 @ Pt.Zero-Only
      Calibr. Extern.     Set to:   0.0
       :       :          Reads:    0.0
      Scanner 9046-395    Ch        T-degC...
       :       :          9         22.86411
       :       :          10        22.50216
       :       :          11        28.83248
       :       :          12       ********
       :       :          13       ********
       :       :          14       ********
       :       :          15       ********
       :       :          16       ********

    Adjustment Summary: LRN 6   Offset       Offset      Range
      Scanner 9046-395    Ch    (orig.)      (final)     (F.S.)
       :       :          9     0.0          22.86411    1350 degC
       :       :          10    0.0          22.50216    1350 degC
       :       :          11    0.0          28.83248    1350 degC
       :       :          12    0.0          99999.0     1350 degC
       :       :          13    0.0          99999.0     1350 degC
       :       :          14    0.0          99999.0     850 degC
       :       :          15    0.0          99999.0     1350 degC
       :       :          16    0.0          99999.0     1350 degC

Adjustment(s) Completed OK.
```

All three reports above show non-ideal offset adjustments in the extreme. There are two major reasons for this. First, the operator prompts were ignored, and necessary removal of transducers (thermocouples) and the shorting of their inputs was not performed in the first report (LRN #4). Also, in all reports, there were some "over-scale" channel readings (********) that caused *very large* offset adjustments as a result. This is because over-scale thermocouple data values are actually 99999 and over-scale resistance data values are actually 10,000,000. Any final offsets at or near these values indicates that the affected channel was over-scale (and probably not configured correctly) before the adjustment began. However, regardless of the bad data, the Zero-adjustment **was a success**. You can prove this by displaying the ***Run*** form for this module after the adjustment, and noticing that all the "adjusted" channels (including the "bad" channels) now read **zero** – even if they read something else, including over-scale (********), before.

The report above does not include any "details", but if the adjustment is repeated with the "Include Details" option specified, the new report would show step-by-step processing steps, as well as any user prompts. See bottom left of the ***'Configure | Calibrators (NUSS)'*** form for setting this option). The following example shows this extra information (for LRN #4 only):

```
--- NUSS 'Calibration Adjustment Report' for a Single Module:
----  9046-395 () Scanner associated with LRN 4,7,6 (Scans/Avg=64 Sfw.Ver.=2.43)
--- Session is adjusting calibration Zero-Only
--- Report generated: 06-29-2007 at 10:35:57
--- Contents include Summary Results and Detailed Events

  Adjusting Offset For 1 Scanner Module using LRN 4 (Extern.) :

                 LRN #4 is assigned to an EXTERNAL Temperature Calibrator.
                  You are PROMPTED to perform special tasks MANUALLY that
              are important to obtaining a successful calibration adjustment!

      Original Offset coeffs. (Scanner 9046-395's Ch 1-3,5-6) BEFORE adjust:
        0.0  0.0  0.0  0.0  0.0

        Scanner 9046-395's  Run/Cal valves do not exist -- prompts may be necessary.
      Operator manually prompted at 10:35:58 AM as follows:
        Press [Resume] when ready to Zero-adjust by either Thermal Bath or Auto-UTR method.
      Operator cleared the prompt at 10:36:00 AM
      Operator manually prompted at 10:36:00 AM as follows:
        DISCONNECT && SHORT-CIRCUIT all TC channels (this LRN) so Auto-UTR values can be
        read. (then press [Resume])
      Operator cleared the prompt at 10:36:01 AM
        Temperature Calibrator Reads Auto-UTR per-chan values in degC after prompt cleared
        UTR Data (Scanner 9046-395's Ch 1) reading is: 28.21719
        Module Data (Scanner 9046-395's Ch 1) reading is: 23.24663
        UTR Data (Scanner 9046-395's Ch 2) reading is: 28.31463
        Module Data (Scanner 9046-395's Ch 2) reading is: 25.66327
        UTR Data (Scanner 9046-395's Ch 3) reading is: 28.35839
        Module Data (Scanner 9046-395's Ch 3) reading is: 28.31758
        UTR Data (Scanner 9046-395's Ch 5) reading is: 28.54739
        Module Data (Scanner 9046-395's Ch 5) reading is: 25.78327
        UTR Data (Scanner 9046-395's Ch 6) reading is: 28.61508
        Module Data (Scanner 9046-395's Ch 6) reading is: 99999.0
        Scanners' Offset coefficients were adjusted.
```

```
      Final Offset coeffs. (Scanner 9046-395's Ch 1-3,5-6) AFTER adjust:
        -4.859321  -2.586042  -0.082896  -2.428998  99970.48

      Indicated Readings: LRN 4 @ Pt.Zero-Only
        Calibr. Extern.     Set to:  Auto-UTR per-ch. values
         :       :          Reads:   Auto-UTR per-ch. values
        Scanner 9046-395    Ch        T-degC...
         :       :          1         23.31762
         :       :          2         25.69077
         :       :          3         28.2082
         :       :          5         26.02717
         :       :          6        ********

      Adjustment Summary: LRN 4   Offset       Offset      Range
        Scanner 9046-395    Ch    (orig.)      (final)     (F.S.)
         :       :          1     0.0         -4.859321    300 degC
         :       :          2     0.0         -2.586042    300 degC
         :       :          3     0.0         -0.082896    1350 degC
         :       :          5     0.0         -2.428998    1350 degC
         :       :          6     0.0          99970.48    1350 degC

      Scanner 9046-395's  Run/Cal valves do not exist -- prompts may be necessary.
    Operator manually prompted at 10:36:06 AM as follows:
      Press [Resume] after adjusted Thermal ports are restored to normal state after Zero
       adjust.
    Operator cleared the prompt at 10:36:07 AM

Adjustment(s) Completed OK.
```

F.5 Full *Test* Report File Examples from Chapter 5

F.5.1 Leak Test Report Examples

Two example reports of the **Leak Test** report (that were once included in **Section 5.2** of **Chapter 5**) have been moved here.

The *first example* is the un-annotated complete text of a **Leak Test** report for a Coordinated Group of four modules sharing two LRNs. It is a complete *main* test report containing both *Summary* data (tables) and *Detailed Events*. Only one of the modules in the group is real. The rest are simulated (serial #0) modules, and one of these (a 9816) had 2 unique ranges of transducer, thus it required two LRNs to “calibrate” it.

```
--- NUSS 'Leak Test Report' for Group * including modules:
----  9016-0 () associated with LRN 1
----  9016-669 () associated with LRN 1
----  9021-0 () associated with LRN 2
----  9816-0 () associated with LRN 1,2
--- Session includes Auto 'Full-circuit' cycle only
--- Report generated: 05-29-2001 at 17:16:03
--- Contents includes Summary Results & includes Detailed Events
--- Test conducted by: q  Comment: q

  Testing Leaks For Scanner Modules using LRN 1 (9034-373) :

    Calibrator (9034-373) for LRN 1 Already Connected Externally

      Connecting Calibrator to Scanner 9016-0's 'Cal' Ports...
        Scanner 9016-0's valves should be connected (9016 has no feedback)

      Connecting Calibrator to Scanner 9016-669's 'Cal' Ports...
        Scanner 9016-669's valves should be connected (9016 has no feedback)

      Connecting Calibrator to Scanner 9816-0's 'Cal' Ports...
        Scanner 9816-0's valves connected OK (verified by feedback)

      ------------------------------------------------------------------------------
      Begin an Auto Leak Test Cycle in 'Full-Circuit' Mode at 5:16:27 PM
      ------------------------------------------------------------------------------

      Setting point 1 to 1 psi with Calibrator...
        Calibrator Set to 1 psi Pressure; Reads  0.999963 psi after 6 sec.stabiliz.delay
        Calibrator's valves set to HOLD its last generated pressure
        Module Data (Scanner 9016-0's Ch 1-16) readings are:
          1.001  1.002  1.003  1.004  1.005  1.006  1.007  1.008
          1.009  1.01  1.011  1.012  1.013  1.014  1.015  1.016

        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          1.000294  1.000288  1.000444  1.000223  1.001467  1.00039  1.000398  1.000484
          1.000464  1.000511  1.000474  1.000418  1.001287

        Module Data (Scanner 9816-0's Ch 1,3,5,7,9,11,13,15) readings are:
          1.001  1.002  1.003  1.004  1.005  1.006  1.007  1.008

        Begin Holding Pressure for 120 Sec. at 5:16:44 PM
          Calibrator Reading  before delay is  1.00002 psi
          Calibrator Reading after delay is  0.999983 psi
```

```
        End Holding Pressure at 5:18:44 PM

        Module Data (Scanner 9016-0's Ch 1-16) readings are:
          1.001  1.002  1.003  1.004  1.005  1.006  1.007  1.008
          1.009  1.01  1.011  1.012  1.013  1.014  1.015  1.016

        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          0.998709  0.998733  0.998887  0.998503  1.001292  0.998782  0.998886  0.998894
          0.998948  0.998847  0.998888  0.99891  1.000035

        Module Data (Scanner 9816-0's Ch 1,3,5,7,9,11,13,15) readings are:
          1.001  1.002  1.003  1.004  1.005  1.006  1.007  1.008

      Setting Zero point to 0 psi with Calibrator...

      Measured Leaks: LRN 1 for Cycle Full-Circuit
        Calibr. 9034-373    Set to: 1.0
         :       :          Reads:  0.999963
                                    Initial      Final        Leak(-)  *out*
        Scanner 9016-0      Ch      Press(psi)   Press(psi)   Err(%FS)  >2%
         :       :          1       1.001        1.001        0.0
         :       :          2       1.002        1.002        0.0
         :       :          3       1.003        1.003        0.0
         :       :          4       1.004        1.004        0.0

              :      examples deleted here for brevity     :

         :       :          13      1.013        1.013        0.0
         :       :          14      1.014        1.014        0.0
         :       :          15      1.015        1.015        0.0
         :       :          16      1.016        1.016        0.0
                                    Initial      Final        Leak(-)  *out*
        Scanner 9016-669    Ch      Press(psi)   Press(psi)   Err(%FS)  >2%
         :       :          1       1.000294     0.998709    -0.158495
         :       :          2       1.000288     0.998733    -0.155503
         :       :          3       1.000444     0.998887    -0.155705
         :       :          5       1.000223     0.998503    -0.172001
         :       :          6       1.001467     1.001292    -0.0175
         :       :          7       1.00039      0.998782    -0.160807
         :       :          9       1.000398     0.998886    -0.151205
         :       :          10      1.000484     0.998894    -0.159001
         :       :          11      1.000464     0.998948    -0.151599
         :       :          12      1.000511     0.998847    -0.166404
         :       :          13      1.000474     0.998888    -0.158596
         :       :          14      1.000418     0.99891     -0.150794
         :       :          16      1.001287     1.000035    -0.125194
                                    Initial      Final        Leak(-)  *out*
        Scanner 9816-0      Ch      Press(psi)   Press(psi)   Err(%FS)  >2%
         :       :          1       1.001        1.001        0.0
         :       :          3       1.002        1.002        0.0
         :       :          5       1.003        1.003        0.0
         :       :          7       1.004        1.004        0.0
         :       :          9       1.005        1.005        0.0
         :       :          11      1.006        1.006        0.0
         :       :          13      1.007        1.007        0.0
         :       :          15      1.008        1.008        0.0

    Setting Scanner 9016-0's 'Run' Ports at End-of-Test...
    Setting Scanner 9016-669's 'Run' Ports at End-of-Test...
    Setting Scanner 9816-0's 'Run' Ports at End-of-Test...

  Testing Leaks For Scanner Modules using LRN 2 (9034-0) :

    Calibrator (9034-0) for LRN 2 Connected Now
```

```
      Scanner 9021-0's  Run/Cal valves non-existent
    Connecting Calibrator to Scanner 9816-0's 'Cal' Ports...
      Scanner 9816-0's valves connected OK (verified by feedback)

    --------------------------------------------------------------------------------
    Begin an Auto Leak Test Cycle in 'Full-Circuit' Mode at 5:17:28 PM
    --------------------------------------------------------------------------------

    Setting point 1 to 5 psi with Calibrator...
      Calibrator Set to 5 psi Pressure; Reads  5.0 psi after 6 sec. stabilization delay
      Calibrator's valves set to HOLD its last generated pressure
      Module Data (Scanner 9021-0's Ch 1-12) readings are:
        5.001  5.002  5.003  5.004  5.005  5.006  5.007  5.008
        5.009  5.01  5.011  5.012

      Module Data (Scanner 9816-0's Ch 2,4,6,8,10,12,14,16) readings are:
        5.001  5.002  5.003  5.004  5.005  5.006  5.007  5.008

      Begin Holding Pressure for 120 Sec. at 5:17:41 PM
        Calibrator Reading  before delay is  5.0 psi
        Calibrator Reading after delay is  5.0 psi
      End Holding Pressure at 5:19:41 PM

      Module Data (Scanner 9021-0's Ch 1-12) readings are:
        5.001  5.002  5.003  5.004  5.005  5.006  5.007  5.008
        5.009  5.01  5.011  5.012

      Module Data (Scanner 9816-0's Ch 2,4,6,8,10,12,14,16) readings are:
        5.001  5.002  5.003  5.004  5.005  5.006  5.007  5.008

    Setting Zero point to 0 psi with Calibrator...

    Measured Leaks: LRN 2 for Cycle Full-Circuit
      Calibr. 9034-0      Set to: 5.0
       :       :          Reads:  5.0
                                  Initial      Final        Leak(-)  *out*
      Scanner 9021-0      Ch      Press(psi)   Press(psi)   Err(%FS)  >2%
       :       :          1       5.001        5.001        0.0
       :       :          2       5.002        5.002        0.0
       :       :          3       5.003        5.003        0.0
       :       :          4       5.004        5.004        0.0

            :       examples deleted here for brevity     :

       :       :          9       5.009        5.009        0.0
       :       :          10      5.01         5.01         0.0
       :       :          11      5.011        5.011        0.0
       :       :          12      5.012        5.012        0.0
                                  Initial      Final        Leak(-)  *out*
      Scanner 9816-0      Ch      Press(psi)   Press(psi)   Err(%FS)  >2%
       :       :          2       5.001        5.001        0.0
       :       :          4       5.002        5.002        0.0
       :       :          6       5.003        5.003        0.0
       :       :          8       5.004        5.004        0.0
       :       :          10      5.005        5.005        0.0
       :       :          12      5.006        5.006        0.0
       :       :          14      5.007        5.007        0.0
       :       :          16      5.008        5.008        0.0

    Scanner 9021-0's  Run/Cal valves non-existent
  Setting Scanner 9816-0's 'Run' Ports at End-of-Test...
  Calibrator (9034-0) for LRN 2 Disconnected

Test(s) Completed OK.
```

This additional example of a **Leak Test** report shows the optional feature of running the test with an Oven temperature (25 ˚C) specified. The “banner” between the report’s header and body show the oven setting conditions. The report’s body has *two additional columns* on the right side for the new *temperature* data and its *out-of-tolerance* flags. These data are flagged for any ±7 degC deviation from the Oven’s *temperature reading*. This example is for a *simulated* module so the flagged errors here were forced and not real errors. This module and its calibrator are also operating in *kPa* pressure units instead of the default *psi* units.

```
--- NUSS 'Leak Test Report' for a Single Module:
----  9816-0 (888) Scanner associated with LRN 1 (Scans/Avg=8 Sfw.Ver.=2.24)
--- Leak Test Mode is Automatic 'Full-circuit' via LRN
--- User MANUALLY Set Leak Check Temperature (25 degC)
--- Report generated: 04-02-2008 at 10:53:02 using NUSS Version: 2.0.13
--- Contents include Summary Results only
--- Test conducted by: John  Comment: Now is the time...

--------------------------------------------------------------------------------
Oven Temp specified for Leak Check (25 deg.C) is expected to be set MANUALLY.
--------------------------------------------------------------------------------

  Testing Leaks For Scanner Module using LRN 1 (9034-0) :

    Set Calibrator to DIFFERENTIAL Mode
      --------------------------------------------------------------------------------
      Begin an Auto Leak Test in 'Auto-Full-Circuit-LRN' Mode at 10:53:11 AM
      --------------------------------------------------------------------------------

        Calibrator's valves set to HOLD its last generated pressure

      Measured Leaks (Duration=120 sec.): LRN 1 for Mode Auto-Full-Circuit-LRN
        Calibr. 9034-0      Set to: 34.0
         :       :          Reads:  34.0
                                    Initial      Final        Leak(-) *out* Initial *out*
        Scanner 9816-0      Ch      Press-kPa(*) Press-kPa(*) Err(%FS)>2.0% Temp-degC>7
         :       :          1       34.001       34.001       0.0     ****  25.001
         :       :          2       34.002       34.002       0.0           25.002
         :       :          3       34.003       34.003       0.0     rise  25.003  ****
         :       :          4       34.004       34.004       0.0           25.004
         :       :          5       34.005       34.005       0.0     ****  25.005
         :       :          6       34.006       34.006       0.0           25.006
         :       :          7       34.007       34.007       0.0           25.007
         :       :          8       34.008       34.008       0.0           25.008
         :       :          9       34.009       34.009       0.0           25.009
         :       :          10      34.01        34.01        0.0           25.01
         :       :          11      34.011       34.011       0.0           25.011
         :       :          12      34.012       34.012       0.0           25.012
         :       :          13      34.013       34.013       0.0           25.013
         :       :          14      34.014       34.014       0.0           25.014
         :       :          15      34.015       34.015       0.0           25.015
         :       :          16      34.016       34.016       0.0           25.016
Test Completed OK.
```

F.5.2 Pressure Accuracy Test Report Examples

Two example reports of the **Pressure Accuracy Test** (that were once included in **Section 5.3** of **Chapter 5**) have been moved here.

This first example is an un-annotated complete text of a **Pressure Accuracy Test** report for a single real module and one LRN. It is a complete *main* test report containing both *Summary* data (tables or sections) and *Detailed Events*. A Re-zero section (table) begins the summary data of the single test run. Note that the column labeled “Cumul.Ofst.Chg.” (and following column showing same data as % of F.S.), are always zero for the first run of a session. The Re-Zero section is followed by several sections (tables) showing how accurately the module responded to the setting of each pressure set point. Any *detailed event data* appears in blocks interleaved around the other sections (tables). Although the module in this example had only three (3) original calibration set points specified, this test program added two other mid-points to make the total number of tested points five (5).

This test also shows only psi and psiA units, but it can show other native units (e.g., kPa and kPaA) if the calibrator and scanner modules are both operating in other like native units.

```
--- NUSS 'Press. Accuracy Test Report' for a Single Module:
----  9016-669 () associated with LRN 1
--- Session includes setting NO temperatures
--- Session includes Rezero Cal. before ea.LRN tested
--- Report generated: 05-24-2001 at 10:58:04
--- Contents includes Summary Results & includes Detailed Events
--- Test conducted by: q  Comment: now is

--------------------------------------------------------------------------------
No Oven Temps are specified.  Default Temp (25 deg.C) assumed already set manually.
--------------------------------------------------------------------------------

  Testing Press. Accuracy For Scanner Module using LRN 1 (9034-373) :

    Calibrator (9034-373) for LRN 1 Already Connected Externally

    ReZero Cal. Optioned Before Each LRN Tested..

      Original Offset coeffs. (Scanner 9016-669's Ch 1-3,5-7,9-14,16) BEFORE adjust:
        0.001699  -0.016193  -0.037635  -0.038601  -0.015001  0.001204 -0.028812 -0.018516  -0.023605
-0.027607  0.000189  -0.000955  -0.378347

      Connecting Calibrator to Scanner 9016-669's 'Cal' Ports...
        Scanner 9016-669's valves should be connected (9016 has no feedback)

      Setting Zero point to 0 psi with Calibrator...
        Calibrator Set to 0 psi Pressure; Reads-0.00001 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          -0.00006  0.000085  -0.000058  -0.000057  0.000038  -0.00001  -0.00001  -0.00001  -0.00001
-0.000029  0.000011  -0.000289  0.00497

      Scanners' Offset coefficients were adjusted.

      Final Offset coeffs. (Scanner 9016-669's Ch 1-3,5-7,9-14,16) AFTER adjust:
        0.001694  -0.016192  -0.037606  -0.038623  -0.012224  0.001234 -0.02883  -0.018465
-0.023562  -0.027627  0.00021  -0.001182  -0.373368

      Adjustment Summary: LRN 1 Offset     Offset     Current    Cumul.     Cumul. *out*
        Scanner 9016-669    Ch  (orig.)    (final)    Zero Rdg.  Ofst.Chg.  Err%FS >Tol  Tol%
          :     :           1   0.001699   0.001694  -0.00006    0.0        0.0         0.25
          :     :           2  -0.016193  -0.016192   0.000085   0.0        0.0         0.25
          :     :           3  -0.037635  -0.037606  -0.000058   0.0        0.0         0.25
          :     :           5  -0.038601  -0.038623  -0.000057   0.0        0.0         0.25
          :     :           6  -0.015001  -0.012224   0.000038   0.0        0.0         0.25
```

```
        :     :           7   0.001204   0.001234  -0.00001    0.0        0.0         0.25
        :     :           9  -0.028812  -0.02883   -0.00001    0.0        0.0         0.25
        :     :           10 -0.018516  -0.018465  -0.00001    0.0        0.0         0.25
        :     :           11 -0.023605  -0.023562  -0.00001    0.0        0.0         0.25
        :     :           12 -0.027607  -0.027627  -0.000029   0.0        0.0         0.25
        :     :           13  0.000189   0.00021    0.000011   0.0        0.0         0.25
        :     :           14 -0.000955  -0.001182  -0.000289   0.0        0.0         0.25
        :     :           16 -0.378347  -0.373368   0.00497    0.0        0.0         0.25

    Data Collection For Accuracy Test Begins..

      Setting point 1 to 0 psi with Calibrator...
        Calibrator Set to 0 psi Pressure; Reads 0.000009 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          -0.00001  0.000038  -0.00001  0.000087  -0.00001  -0.000057  0.000034  -0.00001  -0.000061
-0.00001  -0.000062  0.000041  -0.000009

      Setting point 2 to 0.25 psi with Calibrator...
        Calibrator Set to 0.25 psi Pressure; Reads 0.249915 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          0.250154  0.250081  0.250127  0.250145  0.245434  0.250126  0.250151  0.250031   0.249975
0.250162  0.250058  0.250049  0.258424

      Setting point 3 to 0.5 psi with Calibrator...
        Calibrator Set to 0.5 psi Pressure; Reads 0.499913 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          0.499996  0.500139  0.500083  0.499975  0.492979  0.500031  0.500084  0.500059   0.500039
0.500171  0.499996  0.500035  0.52843

      Setting point 4 to 0.75 psi with Calibrator...
        Calibrator Set to 0.75 psi Pressure; Reads 0.750048 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          0.750049  0.750056  0.749989  0.750021  0.741135  0.749941  0.75  0.749919  0.749981  0.750139
0.749946  0.749983  0.760775

      Setting point 5 to 1 psi with Calibrator...
        Calibrator Set to 1 psi Pressure; Reads 0.999923 psi after 6 sec.stabiliz.delay
        Module Data (Scanner 9016-669's Ch 1-3,5-7,9-14,16) readings are:
          0.999934  0.999922  0.999761  0.999735  0.999285  0.99984  0.999866  0.999835  0.999934  1.000022
0.999954  0.999865  0.975866

      Indicated Readings: LRN 1 @ Pt. 1
        Calibr. 9034-373    Set to: 0.0
         :       :          Reads:  0.000009     Deviation    Deviation *out*
        Scanner 9016-669    Ch       Pr(psid)     Err(psid)    Err(%FS)  >Tol  Tol%
         :       :          1       -0.00001     -0.000019    -0.001895        0.15%
         :       :          2        0.000038     0.000029     0.002905        0.15%
         :       :          3       -0.00001     -0.000019    -0.001895        0.15%
         :       :          5        0.000087     0.000078     0.007805        0.15%
         :       :          6       -0.00001     -0.000019    -0.001895        0.15%
         :       :          7       -0.000057    -0.000066    -0.006595        0.15%
         :       :          9        0.000034     0.000025     0.002505        0.15%
         :       :          10      -0.00001     -0.000019    -0.001895        0.15%
         :       :          11      -0.000061    -0.00007     -0.006995        0.15%
         :       :          12      -0.00001     -0.000019    -0.001895        0.15%
         :       :          13      -0.000062    -0.000071    -0.007095        0.15%
         :       :          14       0.000041     0.000032     0.003205        0.15%
         :       :          16      -0.000009    -0.000018    -0.001795        0.15%

      Indicated Readings: LRN 1 @ Pt. 2
        Calibr. 9034-373    Set to: 0.25
         :       :          Reads:  0.249915     Deviation    Deviation *out*
        Scanner 9016-669    Ch       Pr(psid)     Err(psid)    Err(%FS)  >Tol  Tol%
         :       :          1        0.250154     0.000239     0.023898        0.15%
         :       :          2        0.250081     0.000166     0.0166          0.15%
         :       :          3        0.250127     0.000212     0.021198        0.15%
         :       :          5        0.250145     0.00023      0.022998        0.15%
         :       :          6        0.245434    -0.004481    -0.4481    ****  0.15%
         :       :          7        0.250126     0.000211     0.0211          0.15%
         :       :          9        0.250151     0.000236     0.0236          0.15%
         :       :          10       0.250031     0.000116     0.011599        0.15%
         :       :          11       0.249975     0.00006      0.005999        0.15%
         :       :          12       0.250162     0.000247     0.0247          0.15%
         :       :          13       0.250058     0.000143     0.014299        0.15%
         :       :          14       0.250049     0.000134     0.013399        0.15%
         :       :          16       0.258424     0.008509     0.850901  ****  0.15%

      Indicated Readings: LRN 1 @ Pt. 3
        Calibr. 9034-373    Set to: 0.5
```

```
      :       :        Reads:  0.499913    Deviation    Deviation *out*
    Scanner 9016-669   Ch      Pr(psid)    Err(psid)    Err(%FS)  >Tol  Tol%
      :       :        1       0.499996    0.000083     0.0083          0.15%
      :       :        2       0.500139    0.000226     0.022599        0.15%
      :       :        3       0.500083    0.00017      0.017002        0.15%
      :       :        5       0.499975    0.000062     0.006199        0.15%
      :       :        6       0.492979   -0.006934    -0.693402  ****  0.15%
      :       :        7       0.500031    0.000118     0.011799        0.15%
      :       :        9       0.500084    0.000171     0.017098        0.15%
      :       :        10      0.500059    0.000146     0.0146          0.15%
      :       :        11      0.500039    0.000126     0.012597        0.15%
      :       :        12      0.500171    0.000258     0.0258          0.15%
      :       :        13      0.499996    0.000083     0.0083          0.15%
      :       :        14      0.500035    0.000122     0.012198        0.15%
      :       :        16      0.52843     0.028517     2.851698  ****  0.15%

  Indicated Readings: LRN 1 @ Pt. 4
    Calibr. 9034-373   Set to: 0.75
      :       :        Reads:  0.750048    Deviation    Deviation *out*
    Scanner 9016-669   Ch      Pr(psid)    Err(psid)    Err(%FS)  >Tol  Tol%
      :       :        1       0.750049    0.000001     0.000101        0.15%
      :       :        2       0.750056    0.000008     0.000805        0.15%
      :       :        3       0.749989   -0.000059    -0.005901        0.15%
      :       :        5       0.750021   -0.000027    -0.0027          0.15%
      :       :        6       0.741135   -0.008913    -0.891298  ****  0.15%
      :       :        7       0.749941   -0.000107    -0.010699        0.15%
      :       :        9       0.75       -0.000048    -0.004798        0.15%
      :       :        10      0.749919   -0.000129    -0.012898        0.15%
      :       :        11      0.749981   -0.000067    -0.0067          0.15%
      :       :        12      0.750139    0.000091     0.009102        0.15%
      :       :        13      0.749946   -0.000102    -0.010198        0.15%
      :       :        14      0.749983   -0.000065    -0.006497        0.15%
      :       :        16      0.760775    0.010727     1.072705  ****  0.15%

  Indicated Readings: LRN 1 @ Pt. 5
    Calibr. 9034-373   Set to: 1.0
      :       :        Reads:  0.999923    Deviation    Deviation *out*
    Scanner 9016-669   Ch      Pr(psid)    Err(psid)    Err(%FS)  >Tol  Tol%
      :       :        1       0.999934    0.000011     0.001103        0.15%
      :       :        2       0.999922   -0.000001    -0.000101        0.15%
      :       :        3       0.999761   -0.000162    -0.016201        0.15%
      :       :        5       0.999735   -0.000188    -0.018799        0.15%
      :       :        6       0.999285   -0.000638    -0.063801        0.15%
      :       :        7       0.99984    -0.000083    -0.008297        0.15%
      :       :        9       0.999866   -0.000057    -0.005698        0.15%
      :       :        10      0.999835   -0.000088    -0.008798        0.15%
      :       :        11      0.999934    0.000011     0.001103        0.15%
      :       :        12      1.000022    0.000099     0.009906        0.15%
      :       :        13      0.999954    0.000031     0.003099        0.15%
      :       :        14      0.999865   -0.000058    -0.0058          0.15%
      :       :        16      0.975866   -0.024057    -2.405697  ****  0.15%

 Setting Zero point to 0 psi with Calibrator...
 Setting Scanner 9016-669's 'Run' Ports at End-of-Test...

Test(s) Completed OK.
```

The *per-channel* tolerance values in above report (right-most column) would seem to be unnecessary. However, if the report were for a NetScanner **Model 902x** module, the actual tolerance values do actually vary from *channel to channel*, thus the need for this data column. This is because the **Model 9402** type plug-in transducers have a major accuracy difference (0.5 % F.S.) when compared to other more accurate **Model 9400** and **9401** type transducers (0.05 % F.S.). Older versions of this program did not need to account for such a large (X 10) per-channel tolerance difference. In the older reports the tolerance was shown at the end of the *header* line above the stars (*out*) column as **">0.15%"** instead of the new fixed header **">Tol Tol%"**. When a Group test also generates *separate reports* (for each module in the group) this extra tolerance data column also appears in such reports.

The following additional example of a **Pressure Accuracy Test** report shows the extra temperature data (per scanner channel) shown when the test is run with one or more Oven temperatures specified. In this case only a "manual" Oven setting is used (25 °C). The "banner" between the report's header and body show the oven setting conditions. The first report section shows only the **Cumulative Zero** summary report. All following sections for the **Pressure Accuracy** data have *two additional columns* on the right side for the new *temperature* data and its *out-of-tolerance* flags. These data are flagged for any ±7 degC deviation from the Oven's *temperature reading*. This example is for a *simulated* module so the flagged errors here were forced and thus are not real errors. This module and its calibrator are also operating in *kPa* pressure units instead of the default *psi* units. To shorten the example, data from channels 4-13 are deleted an replaced by "ditto" marks.

```
--- NUSS 'Press. Accuracy Test Report' for a Single Module:
----  9816-0 (888) Scanner associated with LRN 1 (Scans/Avg=8 Sfw.Ver.=2.24)
--- Session includes setting NO temperatures
--- Session includes Rezero Cal.before ea. Differntial LRN tested (skipped if Abs.Mode LRN)
--- Report generated: 04-02-2008 at 11:15:48 using NUSS Version: 2.0.13
--- Contents include Summary Results only
--- Test conducted by: John  Comment: Now is the time...

--------------------------------------------------------------------------------
No Oven Temps are specified.  Default Temp (25 deg.C) assumed already set manually.
--------------------------------------------------------------------------------

 Testing Press. Accuracy For Scanner Module using LRN 1 (9034-0) :

  Set Calibrator to DIFFERENTIAL Mode (and Rezero it)
  ReZero Cal. Optioned Before Each LRN Tested..

    Scanners' Offset coefficients were adjusted.

 Adjustment Summary: LRN 1 Offset  Offset  Current  Cumul.   Cumul.   *out* Cumul.Range
    Scanner 9816-0      Ch  (orig.) (final) ZeroRdg. Ofst.Chg.Err%FS  >0.05% Cnt  (F.S.)
     :       :          1   0.0001  0.0004  0.001   -0.0002  -0.000588       4    34 kPa
     :       :          2   0.0002  0.0005  0.002   -0.0002  -0.600006 ****  4    34 kPa
     :       :          3   0.0003  0.0006  0.003   -0.0002  -0.000588       4    34 kPa
                        :              :       :
     :       :          14  0.0014  0.0017  0.014   -0.0002  -0.000588       4    34 kPa
     :       :          15  0.0015  0.0018  0.015   -0.0002  -0.000588       4    34 kPa
     :       :          16  0.0016  0.0019  0.016   -0.0002  -0.000588       4    34 kPa

 Data Collection For Accuracy Test Begins..

  Indicated Readings: LRN 1 @ Pt. 1
    Calibr. 9034-0  Set to:-34.0 kPa(*)
     :       :      Reads: -34.0         Deviation    Deviation *out*               *out*
    Scanner 9816-0  Ch       Pr-kPa(*)    Err-kPa(*)   Err(%FS)  >Tol    Tol%  Temp-degC>7
     :       :      1       -33.999       0.000999     0.00294         0.05   25.001
     :       :      2       -33.998       0.001999     0.005879        0.05   25.002
     :       :      3       -33.997       0.002998     0.008819        0.05   25.003  ****
                           :            :        :
     :       :      13      -33.987       0.013        0.038237        0.05   25.013
     :       :      14      -33.986       0.014        0.041176        0.05   25.014
     :       :      15      -33.985       0.014999     0.044116        0.05   25.015
     :       :      16      -33.984       0.015999     0.047055        0.05   25.016

  Indicated Readings: LRN 1 @ Pt. 2
    Calibr. 9034-0  Set to:-17.0 kPa(*)
     :       :      Reads: -17.0         Deviation    Deviation *out*               *out*
```

```
    Scanner 9816-0  Ch      Pr-kPa(*)    Err-kPa(*)    Err(%FS)  >Tol   Tol%   Temp-degC>7
     :        :     1      -16.999       0.000999      0.00294          0.05   25.001
     :        :     2      -16.998       0.002001      0.005885         0.05   25.002
     :        :     3      -16.997       0.003         0.008824         0.05   25.003  ****
                           :            :        :
     :        :     14     -16.986       0.014         0.041176         0.05   25.014
     :        :     15     -16.985       0.014999      0.044116         0.05   25.015
     :        :     16     -16.984       0.016001      0.047061         0.05   25.016

  Indicated Readings: LRN 1 @ Pt. 3
    Calibr. 9034-0  Set to: 0.0 kPa(*)
     :        :     Reads:  0.0          Deviation     Deviation *out*                *out*
    Scanner 9816-0  Ch      Pr-kPa(*)    Err-kPa(*)    Err(%FS)  >Tol   Tol%   Temp-degC>7
     :        :     1       0.001        0.001         0.002941         0.05   25.001
     :        :     2       0.002        0.002         0.005882         0.05   25.002
     :        :     3       0.003        0.003         0.008824         0.05   25.003  ****
                           :            :        :
     :        :     14      0.014        0.014         0.041176         0.05   25.014
     :        :     15      0.015        0.015         0.044118         0.05   25.015
     :        :     16      0.016        0.016         0.047059         0.05   25.016

  Indicated Readings: LRN 1 @ Pt. 4
    Calibr. 9034-0  Set to: 17.0 kPa(*)
     :        :     Reads:  17.0         Deviation     Deviation *out*                *out*
    Scanner 9816-0  Ch      Pr-kPa(*)    Err-kPa(*)    Err(%FS)  >Tol   Tol%   Temp-degC>7
     :        :     1       17.001       0.000999      0.00294          0.05   25.001
     :        :     2       17.002       0.002001      0.005885         0.05   25.002
     :        :     3       17.003       0.003         0.008824         0.05   25.003  ****
                           :            :        :
     :        :     14      17.014       0.014         0.041176         0.05   25.014
     :        :     15      17.015       0.014999      0.044116         0.05   25.015
     :        :     16      17.016       0.016001      0.047061         0.05   25.016

  Indicated Readings  LRN 1 @ Pt. 5
    Calibr. 9034-0  Set to: 34.0 kPa(*)
     :        :     Reads:  34.0         Deviation     Deviation *out*                *out*
    Scanner 9816-0  Ch      Pr-kPa(*)    Err-kPa(*)    Err(%FS)  >Tol   Tol%   Temp-degC>7
     :        :     1       34.001       0.000999      0.00294          0.05   25.001
     :        :     2       34.002       0.001999      0.005879         0.05   25.002
     :        :     3       34.003       0.002998      0.008819         0.05   25.003  ****
                           :            :        :
     :        :     14      34.014       0.014         0.041176         0.05   25.014
     :        :     15      34.015       0.014999      0.044116         0.05   25.015
     :        :     16      34.016       0.015999      0.047055         0.05   25.016

Test Completed OK.
```

F.5.3 Leak & Pressure Setting Accuracy Test Report Examples

The complete text of a **Leak & Pressure Setting Accuracy Test** report (that was once included in **Section 5.4** of **Chapter 5**) now appears below:

The next example shows a typical ***Leak & Pressure Setting Accuracy*** test report for a 9034 *calibrator* module. The report's header is followed by a *section* of data for *each test phase* checked when that session was started. If a phase was skipped, a message indicating this fact appears instead for that phase's report section (not shown in this example). In this example all four phases where initially checked and completed. The report sections of the larger phases are heavily edited for brevity using *repetition deleted* lines such as the following:

```
      :              :              :              :
```

Each report normally shows only psi and psiA units, but can show other units (e.g., kPa) if the calibrator module was set to operate in these native units.

```
--- NUSS 'Pressure Setting Accuracy Test Report' for a Single 903x Calibrator Module:
---- 9034-490 () Quartz Calibrator (Freq.=2000 Sfw.Ver.=2.4)
--- Full Scale (FS) Pressure of module: 300 psiA
--- All Phases tested at Max. pressure of 315.000000 psiA
--- Exercise Phase tested From 315.000000 psiA To 30.000000 psiA in 5.700 Steps
--- Hold Phase tested at 25%, 50%, 75% and 100% Max. absolute pressures with 0.015%FS Tol.
--- Coarse Setting Errors over 0.01 %FS are flagged for Exercise phase
--- Fine Setting Errors (per module status) are also verified for all phases
--- Max. Wait Time to set pressure = 60 sec.
--- Report generated: 03-10-2008 at 17:12:27 using NUSS Version: 2.0.12
--- Test conducted by: q  Comment:

  Leak Test Phase Results (Hold Period = 120 sec):

     Pressure     Pressure     Pressure      Leak(-)     *out*
     Setting     Init.-Rdg    Final-Rdg     Err(%FS)    >2.0%

   315.000000   314.984545   314.107695    -0.278365

    Total Test Points in Leak phase:  2
    Erroneous Points in Leak phase:   0

  Exercise Test Phase Results Sample Period = 1 sec; Dwell Period = 1 sec):

     Pressure     Pressure       Error        Error      Elapsed       *out*
     Setting      Reading        (psiA)       (%FS)      (secs)       >0.01%

   315.000000   314.978998    -0.021002    -0.007001       15
    30.000000    29.999670    -0.000330    -0.000110       18
   309.300000   309.278105    -0.021895    -0.007298       12
    35.700000    35.705959     0.005959     0.001986       16
       :              :              :              :
   303.600000   303.593244    -0.006756    -0.002252       15
    35.700000    35.699096    -0.000904    -0.000301       17
   309.300000   309.295572    -0.004428    -0.001476       15
    30.000000    29.999806    -0.000194    -0.000065       17

    Total Test Points in Exercise phase:  101
    Erroneous Points in Exercise phase:   0
    Mean Pressure:      171.085747
    Mean Error:         -0.003362
    Std. Dev. of Error: 0.012900
```

```
Hold Test Phase Results (Hold Period = 100 sec; Dwell Period = 1 sec):

   Pressure     Pressure       Error        Error      Elapsed      *out*
   Setting      Reading        (psiA)       (%FS)      (secs)      >0.015%

  78.750000    78.754787     0.004787     0.001596        5
               78.748008    -0.001992    -0.0006
               78.748008    -0.001992    -0.0006
               78.761565     0.011565     0.0038
     :             :             :             :
               78.755015     0.005015     0.0016
               78.748008    -0.001992    -0.0006
               78.761565     0.011565     0.0038
               78.748008    -0.001992    -0.0006

  Total Test Points for this Hold point:  97
  Erroneous Points for this Hold point:   0

 157.500000   157.498735    -0.001265    -0.000422        8
              157.511851     0.011851     0.0039
              157.505094     0.005094     0.0016
              157.505094     0.005094     0.0016
     :             :             :             :
              157.499133    -0.000867    -0.0002
              157.505094     0.005094     0.0016
              157.498735    -0.001265    -0.0004
              157.498735    -0.001265    -0.0004

  Total Test Points for this Hold point:  97
  Erroneous Points for this Hold point:   0

 236.250000   236.251971     0.001971     0.000657       16
              236.240642    -0.009358    -0.0031
              236.252539     0.002539     0.0008
              236.257920     0.007920     0.0026
     :             :             :             :
              236.252539     0.002539     0.0008
              236.240642    -0.009358    -0.0031
              236.240642    -0.009358    -0.0031
              236.246023    -0.003977    -0.0013

  Total Test Points for this Hold point:  97
  Erroneous Points for this Hold point:   0

 315.000000   315.002674     0.002674     0.000891       21
              315.001938     0.001938     0.0006
              314.990843    -0.009157    -0.0030
              315.008222     0.008222     0.0027
     :             :             :             :
              314.997127    -0.002873    -0.0009
              315.002674     0.002674     0.0008
              315.002674     0.002674     0.0008
              314.997127    -0.002873    -0.0009

  Total Test Points for this Hold point:  97
  Erroneous Points for this Hold point:   0


Overshoot Test Phase Results (Sample Period = 0.2 sec):

   Pressure     Pressure       Error        Error      Setting      *out*
   Setting      Reading        (psiA)       (%FS)      (done)       >2.0%

  30.000000    33.537729     0.000000     0.000000
               18.937393     0.000000     0.000000
 315.000000    15.865039     0.000000     0.000000
```

```
            111.319298  -203.680700   -64.660540
            231.764617   -83.235380   -26.423930
            238.932195   -76.067800   -24.148510
            251.275149   -63.724850   -20.230110
            256.576321   -58.423680   -18.547200
            265.413258   -49.586740   -15.741820
            269.095356   -45.904640   -14.572900
            272.373683   -42.626320   -13.532160
            278.030625   -36.969380   -11.736310
            280.339274   -34.660720   -11.003400
            284.178425   -30.821580    -9.784627
            285.859630   -29.140370    -9.250911
            288.646583   -26.353420    -8.366164
            289.839541   -25.160460    -7.987447
            291.727573   -23.272430    -7.388072
            292.576322   -22.423680    -7.118628
            293.990593   -21.009410    -6.669653
            294.612305   -20.387700    -6.472284
            295.691000   -19.309000    -6.129841
            296.153785   -18.846210    -5.982925
            296.926546   -18.073450    -5.737605
            297.271136   -17.728860    -5.628211
            297.873435   -17.126570    -5.437005
            298.138194   -16.861810    -5.352954
            298.572501   -16.427500    -5.215079
            298.757598   -16.242400    -5.156318
            299.067170   -15.932830    -5.058041
            299.196602   -15.803400    -5.016952
            299.421666   -15.578330    -4.945503
            299.517304   -15.482700    -4.915142
            299.674808   -15.325190    -4.865140
            299.753551   -15.246450    -4.840143
            299.866723   -15.133280    -4.804215
            299.911021   -15.088980    -4.790152
            299.995371   -15.004630    -4.763374
            300.024178   -14.975820    -4.754230
    :             :              :              :
            314.930555    -0.069445    -0.022046
            314.991579    -0.008421    -0.002673
            314.991579    -0.008421    -0.002673
            314.997127    -0.002873    -0.000912
            314.997127    -0.002873    -0.000912      Set
            315.002674     0.002674     0.000849      Set
            314.997127    -0.002873    -0.000912      Set
            314.997850    -0.002150    -0.000683      Set
    :             :              :              :
            315.002674     0.002674     0.000849      Set
            314.997127    -0.002873    -0.000912      Set
            315.002674     0.002674     0.000849      Set
            314.997127    -0.002873    -0.000912      Set

Total Test Points in Overshoot phase:  188
Erroneous Points in Overshoot phase:   0
```

F.5.4 Noise Test Report Examples

Three complete examples of **Noise Test** reports (that were once included in **Section 5.6** of **Chapter 5**) are now listed below:

This example **Noise Test** report is for a pressure scanner.

```
--- NUSS 'Noise Test Report' File for a Single Module:
---- 9116-5161 () Scanner (Scans/Avg=8 Sfw.Ver.=5.05)
--- Specified Tolerance (%FS) = 0.025
--- Report generated: 07-17-2007 at 15:24:12.
--- Test used 100 samples of unaveraged Pressure E.U. Values
--- Test conducted by: John  Comment: Now is the time...

Chan  Gain   Minimum    Maximum    Diff.      Mean       Std.Dev.   S.D.%FS *Out*  FS Range
 1    20      0.0009     0.0092     0.0082     0.0057     0.0020     0.0130         15.0000
 2    20     -0.0031     0.0056     0.0086     0.0010     0.0019     0.0124         15.0000
 3    20     -0.0056     0.0056     0.0112    -0.0006     0.0021     0.0141         15.0000
 4    20     -0.0036     0.0053     0.0088     0.0008     0.0018     0.0120         15.0000
               :       examples deleted here for brevity    :
 12   20     -0.0032     0.0043     0.0076     0.0007     0.0016     0.0110         15.0000
 13   20     -0.0042     0.0056     0.0098     0.0002     0.0021     0.0140         15.0000
 14   20     -0.0042     0.0069     0.0110     0.0006     0.0020     0.0136         15.0000
 15   20     -0.0037     0.0062     0.0098     0.0014     0.0019     0.0130         15.0000
 16   20     -0.0040     0.0078     0.0118     0.0019     0.0022     0.0148         15.0000

Test Completed OK.
```

The Noise Test report above shows (for each transducer (channel) of the module) its *Gain,* and the *Minimum* and *Maximum* values (plus the absolute value of the *Difference* between them) that were acquired during the test. The *Mean* (average across number of collected samples) and *Standard Deviation* values are also calculated and shown. The next calculated datum shown is the *Standard Deviation as a Percent of Full-Scale (S.D.%FS)* in the next-to-last numerical column. That column may be flagged (****) in the normally-blank column headed ***out*** for any channels that are out-of-tolerance. The final column shows the full-scale (FS) range of that channel in its natural units (e.g., psi in this case).

The report's header prefix indicates

- the *Module ID* (model-serial#), *Scans Per Averaged Sample (Scans/Avg)* setting of module before and after (but not during) the test, and module's *Firmware Version*,
- the *Std. Deviation Tolerance* (as %FS Range) used during the test,
- the *Date* and *Time* test was conducted (and report generated), and
- the *# of Samples* and *Scans Per Averaged Sample (Scans/Avg)* at the time the samples were collected (NOTE: the word **unaveraged** is shown when this parameter is set to its default value of 1),
- Finally, the *Test Conductor* and any *Comment* he/she entered before test started.

If the test is incomplete or aborted for any reason, the "Test Completed OK" message at end of report is replaced with a suitable error message.

The following **Noise Test** report is for a Model 9046 *temperature/resistance scanner*. The report header has the same format as the previous *pressure scanner* example, except for some minor wording differences regarding the engineering units. The out-of-tolerance data for channels (1, 4, 8, and 14) in this example were due to:

- Thermocouple channels without sensors plugged in (i.e., open-thermocouples) or
- Resistance or RTD channel being over-scale or not properly configured internally.

Channel 7 (Gain 1) was set to Sensor Type 1 (volts) while all other channels were assigned thermal or resistance sensor types that can be implied by their Gain and FS Range columns. For example, all TC and RTD channels use Gain=90. Resistance channels may use lower gains for certain ranges. A better test would acquire *counts* or *volts* instead of *E. U.* to avoid the range related peculiarities of E. U. temperature and resistance data

```
--- NUSS 'Noise Test Report' File for a Single Module:
---- 9046-395 () Scanner (Scans/Avg=64 Sfw.Ver.=2.5)
--- Specified Tolerance (%FS) = 0.025
--- Report generated: 07-17-2007 at 15:04:55.
--- Test used 100 samples of unaveraged Primary E.U. Values
--- Test conducted by: John  Comment: Now is the time...

Chan  Gain   Minimum   Maximum   Diff.     Mean      Std.Dev.   S.D.%FS *Out*  FS Range
 1    90     *******   *******   *******   *******   ******     ******  ****   600.0000
 2    90     26.7768   27.0385   0.2617    26.9070   0.0610     0.0203         300.0000
 3    90     26.8185   27.1538   0.3353    27.0113   0.0752     0.0056         1350.0000
 4    45     0.0000    *******   *******   *******   ******     ******  ****   500.0000
               :       examples deleted here for brevity     :
 12   90     31.1990   31.4504   0.2514    31.3574   0.0573     0.0042         1350.0000
 13   90     31.1945   31.4459   0.2514    31.3110   0.0640     0.0047         1350.0000
 14   90     *******   *******   *******   *******   ******     ******  ****   850.0000
 15   90     31.1511   31.4443   0.2933    31.2868   0.0678     0.0050         1350.0000
 16   90     31.1114   31.5305   0.4190    31.3008   0.0940     0.0070         1350.0000

Test Completed OK.
```

The following **Noise Test** report is for Model 903x *standard* and *calibrator* module types. It is similar to the other *scanner* reports above but has only one channel of data. Additionally, the *Scans Per Averaged Sample (Scans/Avg)* parameter cannot be altered for these modules – thus that field (on the test form) cannot be edited by the user. Thus, the default setting used to run the test is this unchangeable *averaged* value, instead of 1 (unaveraged) as was used by other scanners..

```
--- NUSS 'Noise Test Report' File for a Single Module:
---- 9038-0 () DPT Calibrator (Scans/Avg=200 Sfw.Ver.=2.4)
--- Specified Tolerance (%FS) = 0.025
--- Report generated: 07-17-2007 at 15:58:22.
--- Test used 100 samples of averaged Press. E.U. Values (200 samples acquired for ea.avg.)
--- Test conducted by: John  Comment: Now is the time...

Chan  Gain   Minimum    Maximum    Diff.     Mean       Std.Dev.   S.D.%FS *Out*  FS Range
 1    1      1234.0000  1234.0000  0.0000    1234.0000  0.0000     0.0000         5.0000

Test Completed OK.
```

F.5.5 Memory Test Report Examples

The following **Memory Test** report is for the EEPROM inside a Model 9116 *pressure scanner*.

```
--- NUSS 'Memory Test Report' File for a Single Module:
---- 9116-5924 () Scanner (Scans/Avg=8 Sfw.Ver.=5.09)
--- Report generated: 05-15-2009 at 10:35:58 using NUSS Version: 2.0.29
--- Test conducted by: John  Comment: Now is the time...
--- Test wrote character 85 to ALL PAGES of EEPROM and compared result after read.

Data was read from ALL pages and checked OK!

Test Completed OK.
```

Except for the heading (whose last line indicates which character is repeated as a data pattern) and the final "Test Completed OK." line, a *normal* test report (like above example) has just one line in the report body indicating its OK result. However, if *any character* in *any page* fails to compare with the data pattern written to that page, extra lines appear indicating which pages failed (as in this example below)..

```
--- NUSS 'Memory Test Report' File for a Single Module:
---- 9116-5924 () Scanner (Scans/Avg=8 Sfw.Ver.=5.09)
--- Report generated: 05-15-2009 at 11:03:16 using NUSS Version: 2.0.29
--- Test conducted by: John  Comment: Now is the time
--- Test wrote character 32 to ALL PAGES of EEPROM and compared result after read.

   Page 0 had 1 BAD characters after read check!   ******
   Page 1 had 2 BAD characters after read check!   ******
   Page 2 had 1 BAD characters after read check!   ******
   Page 3 had 1 BAD characters after read check!   ******
   Page 4 had 1 BAD characters after read check!   ******
   Page 5 had 1 BAD characters after read check!   ******
   Page 10 had 33 BAD characters after read check!   ******
   Page 21 had 3 BAD characters after read check!   ******
   Page 88 had 1 BAD characters after read check!   ******
   Page 127 had 12 BAD characters after read check!   ******

SOME PAGES FAILED THE READ CHECK!  See details above.   ******

Test Completed OK.
```

One final example shows are test that was ended early by clicking the **[Exit]** button during the write loop. This results in a report that only reads and checks the pages actually written. However, the report indicates this as a foreshortened test in its messages.

```
--- NUSS 'Memory Test Report' File for a Single Module:
---- 9116-5924 () Scanner (Scans/Avg=8 Sfw.Ver.=5.09)
--- Report generated: 05-15-2009 at 12:39:24 using NUSS Version: 2.0.29
--- Test conducted by: John  Comment:
--- Test wrote character 32 UP TO PAGE 3 of EEPROM and compared result after read.

Data was read UP TO PAGE 3 and checked OK!  Full EEPROM NOT TESTED UP TO MAX. PAGE 127

Test Completed OK
```