NTNMR Reference Manual
Table of Contents

1 - NTNMR Introduction

1.1

1.1 What is NTNMR?

NTNMR is the instrument control, data acquisition and data processing software program for Tecmag NMR instruments, including Apollo, Discovery, Orion, DSPect, and Odyssey. NTNMR runs on PCs with Microsoft Windows NT 4.0, Windows 2000 (Professional Edition) and Windows XP. NTNMR can be installed for either full instrument control or for data processing alone.

Many of the features of NTNMR, such as menus, graphical controls manipulated with the mouse, keyboard shortcuts, etc., are common to other Windows software packages the user may be familiar with. These features are not described in this manual.

1.2

1.2 What is in this manual?

This manual is a reference manual, which describes in details the operation of the many features of NTNMR. A brief description of the major software components is given in this chapter. Each major software component group is discussed in detail in the following chapters.

1.3

1.3 Overview

When NTNMR is first launched it displays the Main window shown below. Subsequently, additional windows are opened. The most important of these are the Data display, Dashboard, and Pulse Sequence windows, a brief overview of which are given here. Additional windows are used by the Accessories described in chapter 11, as well as the nD display mode (chapter 8) and the Print Layout mode (chapter 9). Numerous "dialog boxes" are also provided for user entry or selection of information, and are described in the appropriate context.

1.3.1 Main window

The main window, shown below, contains the menus and toolbars for the application, and allows the user to create a new data file or open an existing data file for analysis. Normally, this window contains at least one Document window, for the most-recently-used data set. The menus and toolbars conform to the Windows standards. Menus are described in detail in chapter 5, while toolbars are explained in chapter 2. NTNMR data files are opened as document windows inside the main window, as with other Windows applications.
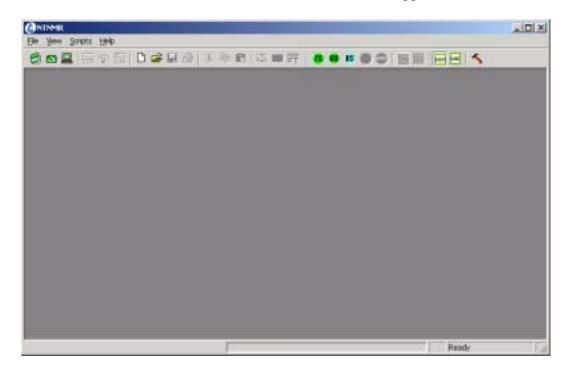
Figure 1.1 NTNMR main window

1.3.2. Data display

The NTNMR document window displays the acquired data at the center. The number of data points in the document is displayed on the top left hand corner of the window. At the top right hand corner, there is a smaller window, which shows the mini spectrum of currently displayed data. The cursor location and the real and imaginary values of the point at the cursor location are displayed on the left of the mini-spectrum window. When the Document window is active all the NTNMR data acquisition, processing and analysis menu options are enabled in the main window and the main tool bar.
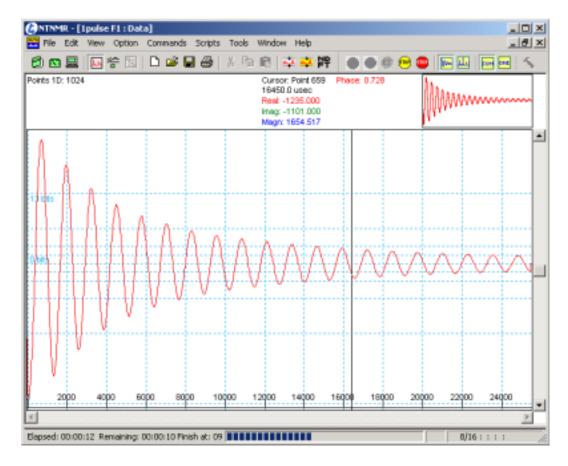
Figure. 1.2 NTNMR data window (document window)

The Document window is described in much more detail in chapter 3.

1.3.3. Dashboard
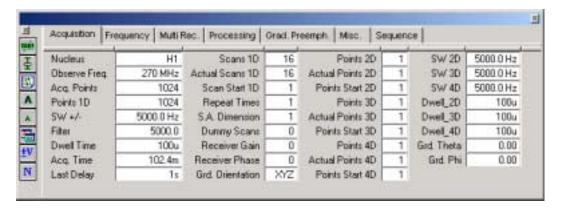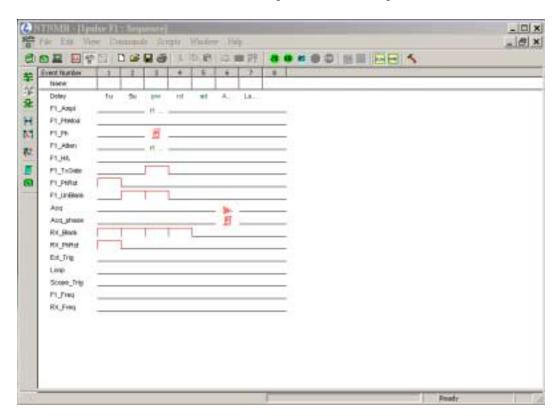
Figure 1.3 Dashboard window

When the Dashboard icon is selected from the main tool bar, the Dashboard window is opened and parameters from the data file will appear. This window displays a categorized list of parameters for acquisition and processing. See chapter 4 for more details on the dashboard.
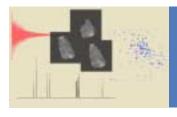
> *The Dashboard window displays the parameters that are read from the data file. Changing the values for parameters displayed in the Dashboard window does not affect the parameters stored with the data file, unless the data is written to a file by using the* **Save** *command.*

1.3.4. Pulse Sequence

Selecting the pulse sequence icon from the main tool bar enables the Pulse Sequence window. The pulse sequence window has a floating tool bar, which, by default is docked to the left of the window. The pulse sequence window is described in much more detail in the chapter 10 on the sequence editor.

Figure 1.4 Pulse Sequence window

2 – Main Window

2.1 Main Toolbar

The Main NTNMR Toolbar (Fig. 2.1) contains an array of buttons that control the most commonly accessed features of NTNMR. Each button on the Main Toolbar is discussed in detail below.

Figure 2.1 Main toolbar

The Main NTNMR Toolbar is "docked" to the top left corner of the main window by default. It can be "undocked" by left clicking anywhere on the Toolbar and dragging. The "undocked" toolbar (Fig. 2.2) can be positioned anywhere on the desktop. Additionally, the Main Toolbar (as well as other Toolbars) can be docked on the bottom of the NTNMR window or on either side. Double-clicking on the Toolbar title bar while it is "undocked" will pop the Toolbar back to its last "docked" position.
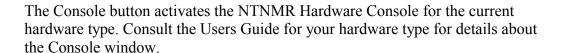
Figure 2.2 The "undocked" main toolbar

The Comment Button

Left-clicking on the Comment button will make the Comment window appear or disappear. The Comment window is a floating, editable, text window containing descriptive information that is stored with each NTNMR data file.

> *Comment Button: Same as **Windows/Comment***

The Console Button

The Console button activates the NTNMR Hardware Console for the current hardware type. Consult the Users Guide for your hardware type for details about the Console window.

> *Console Button: Same as **View/Console Toolbar***

The Dashboard Button

Left-clicking on the Dashboard button will make the Dashboard window appear or disappear. The Dashboard is a "dockable" window that appears at the bottom of the Main Window by default and is always associated with the active data file. When another data file is opened, the contents of the Dashboard window will automatically update to reflect the new data file. See chapter 4 for details about the NTNMR Dashboard.

> *Dashboard Button: Same as **View/Dashboard***

The Data and Pulse Sequence Buttons

These two buttons allow the Pulse Sequence and Data windows to be toggled to the front. Clicking once on the Pulse Sequence button while the data window is displayed causes the associated Pulse Sequence window to appear. Clicking the Data button while the pulse sequence is displayed returns the data to the front. For detailed information about Pulse Sequence display and editing, see chapter 10.

2D Display Button

If a 2D data set is loaded to the front most data window, the 2D Display button will be visible. Clicking on the 2D Display button will activate the 2D Display window. See chapter 8 for more information on 2D display options.

Common Windows Buttons

The next several buttons are common Microsoft Windows buttons:

> *   "New" button: this button opens a new blank NTNMR document.   Same as **File|New**.

*   "Open" button: this button presents the Open File dialog. Same as **File|Open**.

*   "Save" button: when previously unsaved data file is on the front, this button will present the standard "Save As" dialog. In this case, the Save button is identical to **File|Save** and **File|Save As…**. If the data file has already been saved, the Save button simply saves the file, *overwriting the last saved version*. In this case, the Save button is the same as **File|Save**.

*   "Print" button: this button sends the current data file to the currently selected printer. Same as **File|Print…**.

*   "Cut" button: same as **Edit|Cut**.

*   "Copy" button: same as **Edit|Copy**.

*   "Paste" button: same as **Edit|Paste**.

Fit to Window Button

Clicking on the "Fit To Window" button will automatically scale the currently displayed data once without activating Auto Scale mode (see below). If a zoom region is selected, the data in the zoom will be scaled to fit the window. Same as **View | Fit to Window**.

Auto Scale Button

Clicking on this button will activate the Auto Scale mode. In this mode, the data in the document window will always be scaled to fit the window. The scaling is based on the entire data set and not the current zoom. Thus, if Auto Scale is active and a zoom is active, it may appear that the data is not scaled properly. The data is scaled based on the data type that is displayed (i.e. real, imaginary, or magnitude). If more than one type of data is displayed, the following hierarchy is used to determine the data scaling: real, imaginary and magnitude. Therefore, if the real and magnitude data is displayed simultaneously, and the Auto Scale mode is active, the data will be scaled based on the real data only.

The Fit to Window function will always override the Auto Scale function.

nDFT Button

Execute a processing template. [SHIFT]-click to configure the NDFT processing template. See chapter 6 on Commands for detailed information.

2.2 Acquisition Toolbar

2.2

The Acquisition Toolbar (Fig. 2.3) is dockable and behaves the same way as the Main Toolbar; it can be "undocked" and "re-docked" in different locations or "undocked" and allowed to float anywhere on the desktop. Each button of the Acquisition Toolbar is discussed in turn below.

Figure 2.3 The acquisition bar

ZG Button

Zero the signal averager memory and starts an acquisition. Same as **Commands|Acquisition Commands…|Zero and Go and (Ctrl + Z)**.

GO Button

Start an acquisition without zeroing the memory (i.e. continue to acquire). Same as **Commands|Acquisition Commands…|Go and (Ctrl + G)**.

RS Button

Start the Repetitive Scan mode. This command zeroes the memory, begins acquisition and repeats until stopped by the user. Same as **Commands|Acquisition Commands…|Repeat Scan and (Ctrl + R)**. In "Repetitive Scan" mode, only the first scan of a pulse sequence is executed, regardless of the Scans 1D, Points 2D, or other dashboard settings. The entire

pulse sequence is re-compiled between scans, allowing changes of parameters for tune up, etc.

Soft Stop Button

Stop the acquisition after completion of the current receiver phase cycle. Same as **Commands|Acquisition Commands…|Stop and (Ctrl + ESC)**. This button will be grayed unless an acquisition is in progress.

Stop Button

Abort the acquisition immediately. Same as **Commands | Acquisition Commands … | Abort and (ESC)**. This button will be grayed out unless an acquisition is in progress.

Show FID/Show Spectrum Toggle Buttons

These two buttons act as a pair in toggle mode. During acquisition, when the "FID" button (left) is active (i.e. appears 'pushed in' as in the picture above), NTNMR displays time domain data. When the "Spectrum" button (right) is active, NTNMR will display the transformed spectrum (performs BC-FT). In both cases, the display will be updated after every scan. These buttons are used in conjunction with the "SHOW SUM/SHOW ONE" buttons discussed below. More sophisticated processing during acquisition is available via the **Commands | Acquisition Commands …** menu. See chapter 5 for details.

Show Sum/Show One Toggle Buttons

These two buttons act as a pair in toggle mode. During acquisition, when the "SUM" button is active (i.e. appears 'pushed in' as in the picture above), NTNMR will display the accumulated data after each scan. When the "ONE" button is active, NTNMR will show the last single transient acquired and *not* the accumulated data. These buttons are used in conjunction with the "SHOW FID/SHOW SPECTRUM" buttons above.

Reset Button

This button resets the hardware and reloads all default configuration files. This is useful, for example, after the power to the console was turned off temporarily. NTNMR performs a reset automatically when the program starts.

2.3

2.3 Status Bar and Tool Tips

2.3.1 Status Bar

The Status bar appears at the bottom of the Main NTNMR window (*not the data window*) and displays various information about the system.

Figure 2.4 Status Bar

- On the left side of the example above, it shows a description of the NTNMR object that the mouse is currently over. In the case shown in Fig. 2.4, the information displayed is a description of the **ZG** button. Look for information about the various components of NTNMR at this location while using NTNMR.

- In the center of the Status Bar is a "Progress Bar" (shown at approximately half completion in Fig. 2.4) that is updated in real time to show the relative progress of the current acquisition.

- At the far right of the Status Bar is the acquisition counter information. Counters will appear for each dimension in the currently running experiment in the format "*current acq./total acqs. for dimension*". In the example, the current acquisition (in progress) is number 5 out of a total of 12.
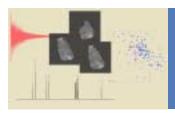
2.3.2 Tool Tips

Many of the tools and buttons in NTNMR (especially on toolbars and in control swap areas) will pop up a Windows style "Tool Tip" when the mouse pointer is passed over them.

Figure 2.5 Tool tip example

The example above shows the Tool Tip for the ZG button on the Acquisition Toolbar.

3 - Document Window

3.1 General Information

3.1

When any NTNMR data file is first opened, an NTNMR Document window appears. An example of a typical 1D data file in the Document window is shown below.

> *The terms "Document window" and "Data window" are synonymous and are used interchangeably throughout the NTNMR manuals.*
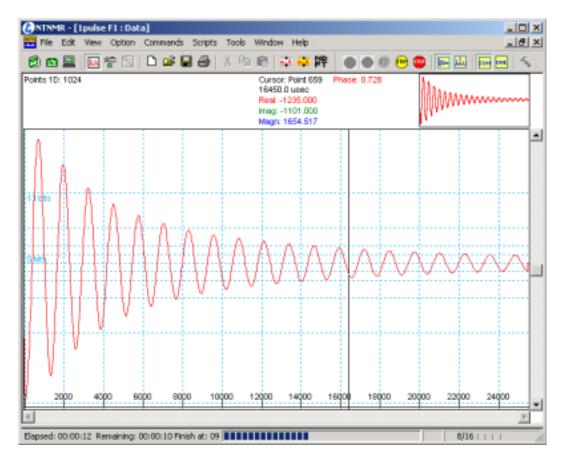
Figure 3.1 The Document window

The Document Window has the following components:

UPPER RIGHT

* Mini Display of NTNMR data

CENTER

* NTNMR Data Area

BOTTOM

- On the left side, a description is displayed of the toolbar icon that the mouse is currently over.

- In the center, is a "Progress Bar" (shown at approximately half completion in the example above) that is updated in real time to show the relative progress of the current acquisition.

- At the far right, is the acquisition counter information. Counters will appear for each dimension in the currently running experiment in the format *current acq./total acqs. for dimension.*

- Status Bar: During the experiment, the display on the left side switches to showing the experiment timing information

The NTNMR Document window can be stretched horizontally or vertically by placing the mouse pointer at the edge of the window, and dragged (double-headed cursor will appear). The window can also be "minimized" by clicking the "Minimize" button in the upper right corner of the window. Alternatively, click the "Maximize" button in the upper right corner of the window to enlarge the NTNMR Document window to full screen.

Another Document window component, the Data Point Information Box, will appear to the left of the Mini Display Area when a data point is selected with the cursor by left-clicking anywhere in the Data portion of the window.

A 2D data set in a Document window is shown below. The 36th record is shown.

> *The term "Record" is used in NTNMR to refer to a 2D, 3D, or 4D point or block of data. This term is also applied to pseudo-2D data such as that from T2 experiments.*
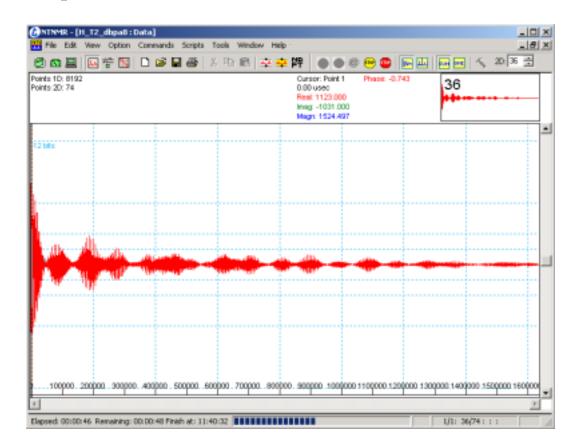
Figure 3.2 A 2D data set in a Document window

3.1.1 Multiple windows

NTNMR can open multiple document windows. **File|Open** will open each NMR data file in a new Data window. The window will be given the same name as the data file, and the window name will be added to the Windows menu list. *Multiple copies of the same data file cannot be opened simultaneously*. If the application requires that the same data set be opened in two separate windows simply copy and rename the data. Available memory is the only limitation on the number of windows that can be opened.

3.2

3.2 NTNMR Document Window Components

3.2.1 NTNMR Data Area

The Data Area displays NTNMR data for FIDs or Spectra. The Data area is resized automatically with the Data window.

3.2.2 Data Points and the Cursor

By default, NTNMR draws the spectrum or FID as a continuous waveform by connecting the individual data points. However, **Points** display can be selected in the **View** menu to display the individual points without connecting lines.

Information about each individual point can be obtained. Select a point by placing a cursor on it: single click the left mouse button in the Data Display Area. The Data Point Information appears above the Data Display Area to the left of the Mini Display Area. The following information about the selected cursor position is displayed:

- point position
- time, ppm, or frequency position
- amplitude of the real, imaginary and magnitude data
- phase

The cursor position can also be changed using the right and left arrow keys. If the [SHIFT] key is held down while using the right and left arrow keys the cursor position will jump by 10 points, otherwise the cursor position will change by one point. The data point information in the data box will automatically be updated to reflect the newly selected point.

3.2.3 Zoom

Zooming horizontally (expanding a region of the FID or spectrum for more detailed examination) is activated by a left mouse button "click and drag" operation over the region of interest in the Data Display Area. When the mouse key is released, a highlighted area in a contrasting color appears. Click the left mouse button in the highlighted region in the Data Display Area to activate the

zoom. The Mini Display Area will be updated with a highlighted region corresponding to the current zoom region displayed in the data display area. Zooming within a zoom is possible by the same procedure. When a zoom area is highlighted in the NTNMR Data Area, clicking outside the highlighted area will cancel the highlight.

Click and drag the left mouse button in the highlighted region of the Mini Display Area to scroll to another area in the data while keeping the same zoom width. To deactivate the zoom, double-click the left mouse button in the Mini Display Area.

To set exact ppm or Hz limits for the zoom parameters, choose **Options/Amplitude Adjustments** and set **F1** and **F2** or set **F2-F1** and **offset** to the desired value. See chapter 5 for more details.

The **View** Menu option **Previous Zoom** allows switching back and forth between two views.

Single clicking at any point within the zoom activates the cursor and causes the Data Point Information to appear. The cursor functions identically in the Zoom as in the Data Display Area described above.

The vertical counterpart of horizontal zoom is controlled in a quite different manner. To expand the data vertically, the arrow keys can be used: up-arrow to increase the vertical expansion, and down-arrow to decrease. In each case the expansion changes by a factor of two for each key press. The vertical offset is controlled by the scroll-bar at the right edge of the data area. Precise control of the vertical zoom is also possible using the **Options|Amplitude Adjustments** mode described in chapter 5.

Scroll bars on the Data Window allow the zoom regions (horizontal and vertical) to be moved without changing their size.

3.2.4 Axis

The Data Axis (displayed below the data in the Data window) can be toggled on and off by selecting **View|Axis**. The Axis Setup dialog is activated by double clicking on the axis in the data display area.
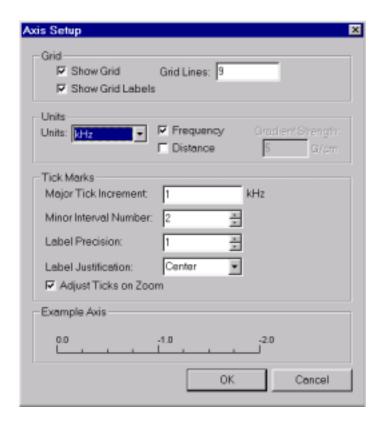
Figure 3.3 The **Axis Setup** window

Axis Setup Controls

Grid

- **Show Grid:** display the horizontal and vertical grids in the spectral display window.

- **Show Grid Labels:** enable grid labels on the vertical axis grid lines

- **Grid Lines:** set the number of vertical axis grid lines to be drawn
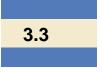
Units

- **Units:** pop-up menu to allow the axis units to be set.

- **Frequency** and **Distance**: allow units to be set to frequency or distance (e.g. for imaging). Active only when the data is in the frequency domain.

- **Gradient Strength**: text field for entering the gradient strength in units of G/cm for calibrating the axis distance units

Tick Marks

- **Major Tick increment:** Increment (set according to the currently selected units) at which major ticks will be drawn on the axis. Major ticks are labeled with a numerical value.

- **Minor Interval Number:** set the number of intervals between labels.

- **Label Precision:** set the number of decimal places of the label.

- **Adjust on Zoom:** this check box enables a scaling of the axis display when the zoom region is selected.
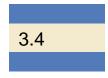
3.3 Data Area Popup Menu

3.3

Right-clicking in the data area will display a small menu of commonly used commands. The selections available in this menu are:

- **Data** will allow you to select which of real, imaginary and amplitude you want displayed similarly to the **View|data** option.
- **Previous Zoom** will only be enabled if you are in zoom mode and will return your display to the previous zoom setting.
- **Acquisition** permits starting an acquisition (**ZG)** in the same way as clicking the ZG button on the acquisition toolbar.
- **Processing|Set Reference.** See the discussion in the next section.
- **Processing|FT** will perform a Fourier transform on the data.
- **Cursor Coordinates** will show the x,y coordinates of the cursor in the data area.

Figure 3.4 The data area Popup menu

3.4 Axis Reference

3.4
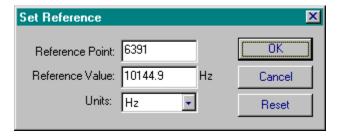
The Set Reference Dialog is displayed when **Reference|Set Reference** is selected from the **View** menu. It can also be displayed by right-clicking on the data area and selecting **Processing|Set Reference.**

Set Reference Dialog Controls
- **Reference Point:** the data point in the spectrum to which the new frequency or PPM value is to be applied

- **Reference Value:** the new value to be assigned to the **Reference Point** entered

- **Units:** the units for the Reference Value

- **OK** will reference the current data set at the Reference Point to the Reference Value. The current cursor position and its value for the currently selected units are loaded by default. These values can be changed directly by typing in the associated text fields. The units can be selected from the Units pop-up.

- **Cancel** will remove the dialog with no change in reference

- **Reset** will re-position zero Hertz in the center of the frequency axis
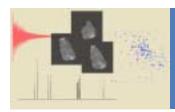
Figure 3.5 The **Set Reference** dialog box

Setting the Spectrum Reference

When the Set Reference dialog is open, the current cursor position is displayed in the **Reference Point** window. If a different point is desired, a new **Reference Point** number can be entered directly. Entering a new **Reference Value** will assign the value entered to the point in the spectrum indicated by the **Reference Point**.

An option for setting an "absolute reference" that will be available to other windows is also available. This option is discussed in detail in the section on the **View** menu.

4 – The Dashboard

The NTNMR Dashboard displays the parameter values stored in the data file, and allows the user to modify them for a new experiment.

4.1 Displaying the Dashboard

4.1
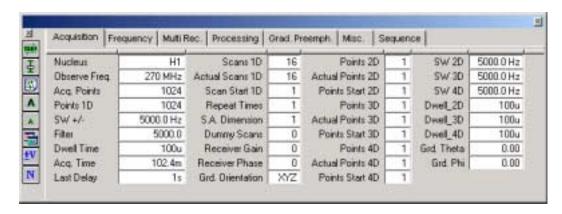
Figure 4.1 The Dashboard.

When the **Dashboard** button is activated and the Dashboard window is opened, parameters from the data file will appear. This window displays a categorized list of parameters for acquisition and processing.

> *The Dashboard window displays the parameters that are read from the data file. Changing the values of any parameters displayed in the Dashboard window does not affect the parameters stored with the data file, unless the data is written to a file by using the **Save** command.*

4.2 Working with the Dashboard

4.2

The dashboard contains several tools that facilitate quick access to information available as well as manipulation of the appearance of the Dashboard.

4.2.1 Re-sizing the Dashboard

The Dashboard is a "dockable" window. In the "docked" state, it can be resized vertically by dragging the dash-bar. In the "undocked" state, the mouse pointer changes while hovering over the Dashboard window edges indicating that it can be resized by dragging the mouse pointer while holding the left mouse button down.

If there are too many parameters to display, a scroll bar will be activated at the bottom of the window to allow scrolling to the parameters out of the current view.

4.2.2 Tabs

The parameters in the dashboard are organized under several categories that appear as a row of tabs along the top of the dashboard. Left-clicking on any tab will update the display to show all of the parameters in that category. Note that not all parameters are editable. Some parameters such as "Actual Scans 1D" are provided for informational purposes and are not editable.

The Tab Button:

Clicking on the **Tab** button will hide the category tabs along the top of the Dashboard.

Real Time Adjustment Mode Button:

This button activates the Real Time Adjustment Mode for interactively adjusting experimental parameters. See chapter 10 on Pulse Sequences for more information.

The Scroll Button:

The **Scroll** button can be used to page through the various Dashboard parameter display categories. Each left-click will cause the display to jump to the next tab and update the display. This button is functional regardless of whether the tabs are currently visible.

The Text Size Buttons:
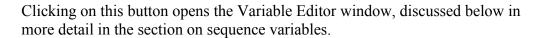
The **Test Size** buttons allow the size of the Dashboard text to be increased and decreased. Left-clicking either size button will cause the display to be refreshed with an increase or decrease of 1 point in the font size.

The Dashboard Customize Button:

Clicking on this button allows the user to customize the dashboard display. See the section below for more details.

Variable Editor Button:

Clicking on this button opens the Variable Editor window, discussed below in more detail in the section on sequence variables.
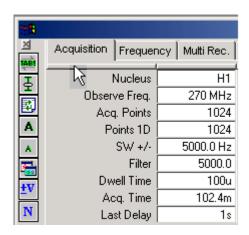
Dashboard View Toggle button: (Value mode) (Formula mode)

See section 4.5.5

4.2.3 The Justification Bars

Left-clicking on the justification bars above the parameter descriptions and values will alternately change the justification from left to right. The white arrow points to the justification bar on the diagram below.

Figure 4.2 The Justification bars

When the Dashboard window is active, use the [TAB] key to scroll through the list, or select a field by left-clicking and hit [RETURN] or [ENTER] to complete any entry.

4.2.4 Multiple Files

There is only one Dashboard window. When multiple files are open, the Dashboard displays the parameters for the dataset associated with the "active" (front-most) window. Clicking on another window will cause the dashboard to update and display the parameters for that window.

4.3

4.3 Customizing the Dashboard

The Dashboard Customize Button:

Clicking on the button will launch the Dashboard Customize applet that allows the user to choose from a pre-saved list of dashboard configurations, load a pre-saved dashboard configuration from disk, or to customize and save a new dashboard configuration.
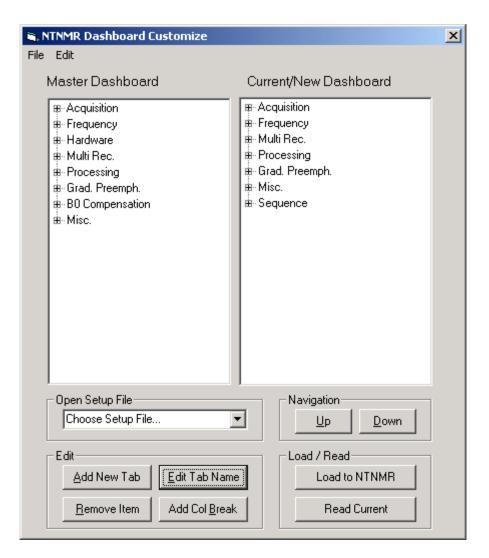
Figure 4.3 Dashboard Customize applet

Most of the features of this window are self-explanatory. To place a parameter in the Current/New Dashboard, select the tab on the right-hand side where the parameter is to appear. Then double-click the parameter name in the left-hand column. It will be added at the bottom of the list for the highlighted tab on the right. The position can then be changed with the Up/Down buttons.

The changes do not take affect until "Load to NTNMR" is clicked.

4.4

4.4 Dashboard Parameters

4.4.1 Master Parameters

*The following codes are used to indicate the functionality of dashboard parameters:

- 1 = editable by the user
- 2 = not editable by the user
- 3 = information field only, no hardware significance
- 4 = may not be functional on all hardware types

The table shown below lists all of the available dashboard parameters in NTNMR. Keep in mind that the dashboard that appears in NTNMR will most likely not display all of the parameters listed and that parameters may be stored (and even duplicated) under any tab. Custom tabs with unique names can also be created.

Dashboard Parameters			
Parameter	**Function**	**Default Tab**	**Info***
Date	Display the NTNMR file creation date	Misc.	1,3
Magnet Field	Magnetic field in Tesla	Misc.	1,3
F1 Ref. Freq.		Misc.	1
Absolute Freq.		Misc.	2
Exp. Start Time	Time the current experiment was started	Misc.	2
Exp. Finish Time	Estimated time the current experiment will finish	Misc.	2
Exp. Elapsed Time	Estimated time the current experiment has been running	Misc.	2
Receiver Gain	Receiver gain	Hardware	1
Receiver Gain Ch1	Channel 1 receiver gain	Hardware	1, 4
Receiver Gain Ch2	Channel 2 receiver gain	Hardware	1, 4
DEC Attn	Parameter for the de-coupler power	Hardware	1, 4

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
Shim Units	The sum of the magnitude calculations for all points, used as a shimming rating	Hardware	2
Observe Freq.	Observe frequency in MHz determined by the Observe Ch.	Frequency	1
Observe Ch.	Assign which transmitter channel is assigned for 'observe' (1-4)	Frequency	1
F1 Freq.	F1 frequency in MHz (F1 base + F1 offset)	Frequency	1
F1 Base	Base observe frequency in MHz	Frequency	1
F1 Offset	Added observe frequency in kHz	Frequency	1
Fn Freq.	Fn frequency in MHz (Fn base + Fn offset)	Frequency	1, 4
Fn Base	Base Fn frequency in MHz	Frequency	1, 4
Fn Offset	Added Fn frequency in kHz	Frequency	1, 4
Nucleus	Display the observation nucleus	Acquisition	3
Points 1D	Number of complex data points (input is adjusted to 2n), any change to this will change Acquisition Points if the Hardware Preference is on for tracking	Acquisition	
Acq. Points	number of complex data points actually sampled (not necessarily 2n, but must be less than or equal to Points 1D), any change will not change Points 1D	Acquisition	1
SW +/-	spectral width in Hz, +/- (editing SW+/- will also change the Dwell Time)	Acquisition	1
Filter	audio filter setting	Acquisition	1, 4
Dwell Time	time per complex point in seconds given by (2*SW+/-)-1 (editing Dwell Time will also change the SW+/-)	Acquisition	1

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
Acq. Time	Display the Points 1D * Dwell Time	Acquisition	2
Scans 1D	Number of averages/accumulations requested	Acquisition	1
Actual Scans 1D	Number of averages/accumulations actually completed	Acquisition	2
Dummy Scans	Number of scans without data averaging before acquisition starts	Acquisition	1
Scan Start 1D	Set the starting scan number (1D)	Acquisition	1
Last Delay	The delay of the last event in a pulse sequence (entered in as a number followed by a unit), used to set the delay between scans	Acquisition	1
SW nD	+/- Spectral width for the n dimension (where n can equal 2, 3, or 4).	Acquisition	1
Dwell_nD	Dwell time for the n dimension	Acquisition	1
Points nD	Number of data points in the n dimension	Acquisition	1
Actual Points nD	Number of points actually collected in the n dimension	Acquisition	2
Points Start nD	Starting point for the n dimension	Acquisition	1
Repeat Times	Number of times to repeat the entire experiment. Used for nD experiments only. Data from each dimension will be added Repeat Times value number of times	Acquisition	1
S.A. Dimension	Set the dimension (1-4) in which the signal averager will accumulate data- see the pulse sequence chapter for more information.	Acquisition	1
S.A. Mode	Set behavior of the Signal averager for the dimension specified in S.A.Dimension- see the pulse sequence chapter for more information.	Acquisition	1
DC.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
DC.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
DC.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A0.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A0.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A0.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A1.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A1.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A1.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A2.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A2.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
A3.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A3.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A3.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A4.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A4.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A4.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A5.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A5.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
A5.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T1.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T1.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T1.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T2.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T2.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T2.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T3.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T3.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T3.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T4.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T4.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T4.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T5.x	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
T5.y	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
T5.z	Gradient pre-emphasis parameter	Grd. Preemph.	1, 4
DC.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
DC.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
DC.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A0.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A0.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A0.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A1.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A1.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A1.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A2.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A2.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A2.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A3.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A3.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A3.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A4.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A4.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A4.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A5.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A5.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
A5.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T1.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T2.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T2.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T2.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T3.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T4.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T5.bx	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T5.by	Gradient pre-emphasis parameter	B0 Compensation	1, 4
T5.bz	Gradient pre-emphasis parameter	B0 Compensation	1, 4

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
Obs. Freq. Rec1	Observed Frequency for Ch1 Receiver	Multi Rec.	1, 4
Obs. Ch. Rec1	Observed channel for Ch1 Receiver	Multi Rec.	1, 4
Rec. Gain Ch1	Gain of Ch1 Receiver	Multi Rec.	1, 4
Rec. Phase Ch1	Phase of Ch1 Receiver	Multi Rec.	1, 4
Acq. Points Rec1	Acquisition point of Ch1 Receiver	Multi Rec.	1, 4
SW +/- Rec1	Spectral width of Ch1 Receiver	Multi Rec.	1, 4
Filter Rec1	Audio filter setting for Ch1 Receiver	Multi Rec.	1, 4
Dwell Time Rec1	Dwell time for Ch1 Receiver	Multi Rec.	1, 4
Acq. Time Rec1	Acquisition time in Ch1 Receiver	Multi Rec.	1, 4
Obs. Freq. Rec2	Observed Frequency for Ch2 Receiver	Multi Rec.	1, 4
Obs. Ch. Rec2	Observed channel for Ch2 Receiver	Multi Rec.	1, 4
Rec. Gain Ch2	Gain of Ch2 Receiver	Multi Rec.	1, 4
Rec. Phase Ch2	Phase of Ch2 Receiver	Multi Rec.	1, 4
Acq. Points Rec2	Acquisition point of Ch2 Receiver	Multi Rec.	1, 4
SW +/- Rec2	Spectral width of Ch2 Receiver	Multi Rec.	1, 4
Filter Rec2	Audio filter setting for Ch2 Receiver	Multi Rec.	1, 4
Dwell Time Rec2	dwell time for Ch2 Receiver	Multi Rec.	1, 4
Acq. Time Rec2	Observed Frequency for Ch2 Receiver	Multi Rec.	1, 4
Shift # Pts.	Points value used by left/right shift Command	Processing	1
LB nD	Line broadening constant in Hz used during exponential multiplication	Processing	1
GB nD	Line broadening constant in Hz used during Gaussian multiplication	Processing	1
DM nD	Line broadening constant in Hz used for double exponential multiplication	Processing	1
SB Shift nD	Phase shift for sin bell multiplication	Processing	1
SB Width nD	Window points for sin bell multiplication	Processing	1
SB Skew nD	Skew factor for sin bell multiplication	Processing	1
TZ 1-4 nD	Points used for a trapezoidal apodization	Processing	1
Traf nD	Estimated T2 used during Traficante-Ziessow apodization	Processing	1
Sys. Phase 0 nD	Zeroth order phase value stored in the system	Processing	1
Sys. Phase 1 nD	First order phase value stored in the system	Processing	1
Phase 0 nD	Zeroth order phase value	Processing	1

Dashboard Parameters cont.			
Parameter	**Function**	**Default Tab**	**Info***
Phase 1 nD	First order phase value	Processing	1
Echo Center nD	Center of echo used in Echo Zero Fill, Echo FFT, and other Echo commands	Processing	1

4.4.2 Sequence Parameters

An additional tab labeled **Sequence** will often be visible. This tab contains user definable Sequence Parameters. If a sequence file is loaded that contains variables, the sequence tab will appear in the dashboard. Consult chapter 10 on Pulse Sequences for more information.

Dashboard parameters that represent time or frequency must be followed by a unit (such as "n" for nanoseconds, "u" for microseconds, "m" for milliseconds and "s" for seconds). Scientific notation, like 1e-5, is also allowed.

> *Sweep width (SW +\-) will be updated after each Dwell Time Character is entered. Acquisition Time will be updated for any change in Acquisition Points, Dwell Time or SW +\-.*

4.5

4.5 Variables and Mathematical Expressions

4.5.1 Variable Types

In addition to the dashboard parameters, NTNMR recognizes the following variable types:

Global variables are not tied to a particular data file or pulse sequence. Global variables are created at the highest level and can be used in any data or sequence file. A global variable will retain its value across all data or sequence files. Global variables are created with the Variable Editor.

Local variables are created and maintained on the sequence or data file level. The scope of a local variable is restricted to the data file or sequence file where it was created. A local variable appears only in the dashboard, and is not explicitly contained in the pulse sequence timing diagram. Local variables are created with the Variable Editor.
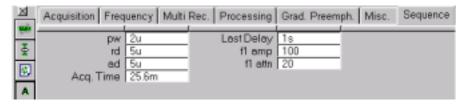
A **sequence variable** is a variable that is assigned in the pulse sequence timing diagram to a particular function in particular event. The most common examples are pulse amplitudes, event times (called 'delays' in the pulse sequence timing diagram), and gradient amplitudes. The scope of a sequence variable is restricted to the data file or sequence file where it was created. Refer to chapter 10 for creation of sequence variables.

Table variables are used in the definition of specific types of tables for which NTNMR has the capacity to automatically calculate the contents. There are currently four types of table where this can be done – delay, gradient amplitude, and frequency. When activating the 'auto' mode for a table, the user has the option to specify values used in the automatic calculation of the table. Values can be hard coded or assigned to a 'table variable' that will then appear in the dashboard for more convenient manipulation. See the section on tables for more information. The scope of a table variable is restricted to the data file or sequence file where it was created. Table variables are created in the respective Table Editors.

4.5.2 Sequence Variables

NTNMR allows *unlimited* sequence variables to be defined. See chapter 10 (Pulse Sequences) for a description of adding sequence variables.

The **Sequence** tab in the Dashboard will immediately be updated to include the newly created variable.

Figure 4.4 An example of the Sequence tab

This allows sequences to be written that can be completely controlled from the dashboard without the user having to edit the sequence directly. Sequence variables and values are saved with the **Data** file or separately in a **Parameter** file. Sequence variables are also saved with **Sequence** files.

Variables can be used multiple times in the sequence. When the sequence encounters a variable name that has already been used the sequence assumes that the same value is to be used for multiple events.

4.5.3 Local and Global Variables

Local and Global variables are created using the Variable Editor. Click the

Variable Editor Button on the dashboard to open the **Variable Editor**
window. This has the same effect as selecting the **View | Variable Editor** menu
item from the main window.

Local variables are displayed on the **Sequence** tab of the dashboard, using blue
text. Global Variables are displayed on the Global Variable tab of the
dashboard.

4.5.4 Variable Editor

The Variable editor, shown below, allows variables to be defined (Local or
Global) or examined (Table and Sequence). In addition, maximum and
minimum limits can be set for variables, for example to avoid accidental entry
of dangerous or ridiculous values.
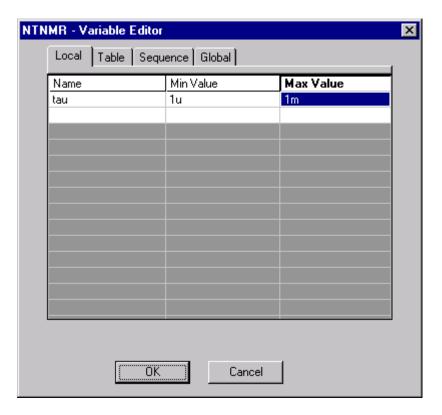
Figure 4.5 The **Variable Editor** applet (local tab)

When editing global variables as shown below, additional buttons appear:
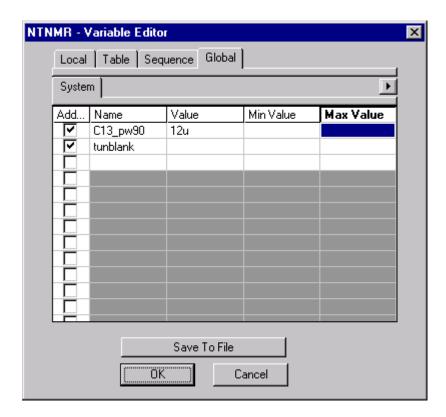
Figure 4.6 The **Variable Editor** applet (global tab)

The **variable set management** button ▶ displays an additional menu, which allows sets of global variables to be loaded or saved. These sets are saved in the directory \ntnmr\setup\username (where username is the Windows user login name). This is useful if the console is used with different probes, magnets, or gradient coils, to load calibration information specific to the experimental setup.

4.5.5 Mathematical expressions

Mathematical expressions can be used in the **sequence** tab of the dashboard only. They define one parameter as a function of other parameters (including local and global variables). This allows one to assign, for example, a pulse width and then cause other pulse widths to be determined by the value of the first. For example if a sequence has a sequence variable called "p90" and another called "p180", the user can set p90 = 5u and set p180 = 2 * p90. Essentially unlimited flexibility is available for the expressions used and many types of mathematical manipulations are supported. The following section details the syntax and illustrates some examples.

Any variable contained in an expression must be encapsulated in the [ ] characters. For example, in the following expression, both *p180* and *p90* are variables.

p180 = [p90] * 2

Calculations are performed in the time base of microseconds (u). For example:

5m + 500u

is evaluated as 5500u.

Expressions that contain values without units and values with units will result in a value with units of microseconds (u), where the unit-less values are assumed to be in units of microseconds.

2 * 5m is evaluated as 10000u (which is equal to 10m)
2 * 5u is evaluated as 10u
2 + 5m is evaluated as 5002u
2 + 5u is evaluated as 7u

Expressions where all values contained are unit-less will return a value with no unit.

2 + 5 is evaluated as 7
2 * 5 is evaluated as 10

The "val" function

NTNMR provides the special function "val" for returning the value portion of any expression. This function removes the units. Keep in mind that the time base for all calculations is microsecond (u).

val(2 * 5m) is evaluated as 10000
val(2 * 5u) is evaluated as 10
val(2 + 5m) is evaluated as 5002
val(2 + 5u) is evaluated as 7

The val function can also be used as part of a more complicated expression.

2.3 * val([tramp] + 1/[pw])

Dashboard View Toggle button: ⬜ (Value mode) ⬜ (Formula mode)

Clicking on this button toggles the Dashboard between the "Value" and "Formula" mode. By default, the Dashboard is in the Value mode. In the Value mode the Dashboard parameters will display the contents as entered by the user, regardless of whether they are numerical values, other variables or mathematical expressions. Whereas in the Formula mode, if the value of a parameter is set to another Dashboard parameter or has a mathematical expression, its numerical value is displayed.

4.5.6 Example

A portion of a dashboard showing some mathematical expressions for calculation of gradient amplitudes (sequence variables Gr, Grr, etc.) from system information (Hz/mm fs) and user entries (Te, offset_r_mm, FOVmm_r, etc.) is shown below in the Formula mode:

Misc.	Sequence	Global Variables		
f_p	=[Gp]*[offset_p_mm]*[Hz/mm fs]/100	Te	40m
G_gauss_spoil	1.5	offset_r_mm	5
Gbal	1.5	offset_p_mm	10
Gr	=(2500/([Hz/mm fs]*[FOVmm_r]))*100	FOVmm_p	10
Gtm	9	FOVmm_r	5
Grr	=[Gr]*0.5*val([tsel]+[tramp])/([trephase]+[tramp])	Tm	50m
Gpr	=[Gp]*0.5*val([tsel]+[tramp])/([trephase]+[tramp])	FOVmm_s	5
Gp	=(2500/([Hz/mm fs]*[FOVmm_p]))*100	Hz/mm fs	1700
Gs	=(2500/([Hz/mm fs]*[FOVmm_s]))*100	Tr	1s
Gsr	=[Gs]*0.5*val([tsel]+[tramp])/([trephase]+[tramp])	offset_s_mm	5
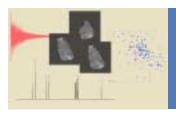
Figure 4.7 The dashboard showing mathematical expresssions

The same dashboard in Values mode looks like this:

Figure 4.8 The dashboard showing the values corresponding to the mathematical expressions displayed in figure 4.7

Note that the system information (Hz/mm fs) could have been a global variable, but was not in this case.

5 - NTNMR Menus

The following is a description of all the pull down menus present above the tool bar (Figure 5.1). With a left click of the mouse on each title on the bar, the menu will be pulled down.

| PITB | File | Edit | View | Option | Commands | Scripts | Tools | Window | Help |
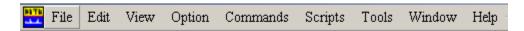
Figure 5.1 Tool Bar

5.1

5.1 "File" Menu

New	Ctrl+N
Open...	Ctrl+O
Revert to Saved	
Close	
Save	Ctrl+S
Save As...	
Import	▶
Export...	
Parameters	▶
Print Setup...	
Print Layout...	
Add to Print Layout	
Print...	Ctrl+P
1 D:\denis\data\loopback_1ch.tnt	
2 pulse shape test.tnt	
3 D:\denis\...\preamphasis.tnt	
4 loopback_1ch_1f1.tnt	
Exit	

Figure 5.2 File pull down menu

Menu items description:

Menu Item	Description
New (Crtl+N)	Create a new blank data window and a blank sequence.
Open (Ctrl+O)	Open a Data file and its associated Parameters in a new window. NTNMR will also open MacNMR files directly if the files are saved with the '*.mn5' file extension.
Revert to Saved	Re-open the last saved copy of the current data file.
Close	Close the currently active window and disposes of the parameters and data.
Import	Allow data in the NMR Pipe format to be imported. Note that the data file will be placed in the currently active window.
Save (Ctrl+S)	Save the current Data and Parameters, overwriting any other Data previously saved under the same file name. The Sequence and Sequence Parameters are also saved. The user is not queried.
Save As...	Save the current Data and Parameters in a user selected file name. The Sequence and Sequence Parameters are also saved. The window name is changed to the new name.
Export…	The user is prompted for a filename in which to save the current data set. Export file types available are Text (*.txt) and MacNMR (*.mn5). If the file is a 2D data set, all of the records will be saved.
Parameters	Display a sub- menu which is described at the end of this section.
Print Setup...	Select the page setup for the printer. (See the section on printing)
Print Layout…	Start the Print Layout utility which is described in detail in chapter 9.
Add to Print Layout	Add current data to open print layout.
Print	Print the current data set.
Recent Files List	NTNMR keeps track of the most recently saved files allowing direct access to the files in the Recent Files List. (See example menu above)
Exit	Quit the NTNMR application.

When a Sequence window is active the following options are added to the File menu:

New Sequence	Replace the current sequence with a blank sequence.
Open Sequence	Select and loads a new Sequence and its Parameters from disk.
Save Sequence	Save the current Sequence and its Parameters to disk.
Save Sequence As	Save the current Sequence and its Parameters to disk under a user specified file name.

<table>
<tr><td>**5.2**</td></tr>
</table>

5.2 "Parameters" Menu

Load Parameters ...
Save Parameters ...

Load Dashboard Setup ...
Save Dashboard Setup ...
Save as Default Dashboard Setup
Reset to Default Dashboard

Figure 5.3 Parameter pull down menu

Menu items description:

Load Parameters…	Load a set of experiment parameters from disk, including sequence parameters.
Save Parameters…	Save the current experiment parameters into a text file. The text file can be viewed and printed with any text editor. Sequence parameters are also saved.
Load Dashboard Setup	Allow the user to select from a set of Dashboard setup.
Save Dashboard Setup	Allow the user to save a customized dashboard setup.
Save as Default Dashboard Setup	Save the present Dashboard setting as default, otherwise, the next time NTNMR is started it will revert back to the original setting.
Reset to Default Dashboard	Set the dashboard to some factory default dashboard.

5.3

5.3 "Edit" Menu

Figure 5.4 Edit pull down menu

Menu items description:

Menu Item	Description
Undo	Inactive
Cut	Cut selected section (for pulse sequence window)
Copy	Copy selected section (for pulse sequence window)
Paste	Paste selected section (for pulse sequence window)
Clear	Inactive
Select All	Select all of the current data (functional for the NTNMR Document Window).
Show Clipboard	Inactive
Setup|Axis	Open the axis setup dialog described in chapter 3.
Preferences	Set defaults for the user. See the Preferences section at the end of this chapter.

5.4 "View" Menu

5.4
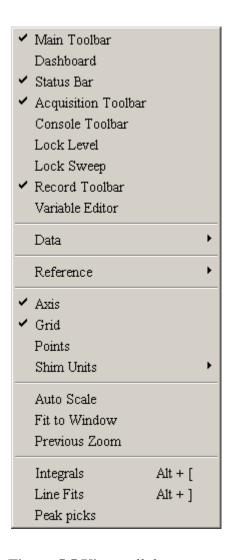
Figure 5.5 View pull down menu

Menu Items description

Menu Item	Description
Main Toolbar	Activate/deactivate the standard NTNMR toolbar.
Dashboard	Activate/deactivate the dashboard.
Status Bar	Activate/deactivate the status bar at the bottom of the NTNMR Main window.
Acquisition Toolbar	Activate/deactivate the Acquisition toolbar.
Console Toolbar	Activate/deactivate the Console toolbar. (Hardware dependent)
Lock Level	Activate the lock level Display.
Lock Sweep	Activate the floating window for the lock display (hardware dependent).
Record toolbar	Activate/deactivate the Record toolbar (ND experiments).
Variable Editor	Activate the floating window for the Variable Editor
Data|Real (Ctrl + R)	Display the real part of complex data.
Data|Imaginary (Ctrl + I)	Display the imaginary part of complex data.
Data|Magnitude (Ctrl + M)	Display the power spectrum of complex data.

Turning off all three Data Display modes will cause only the grid and axes to appear in the display area. If you turn off the grid and axes displays, the program will not appear to be functioning.

Reference|Set Reference	Open the Set Reference Dialog.
Reference|Get Absolute Reference	Set the reference mark for the current spectrum to the Absolute Reference that was saved with Set Absolute Reference.
Reference|Set Absolute Reference	Save the current reference mark for later use. This function allows a reference mark to be applied to another data window.
Axis	Draw a time or frequency axis in the NTNMR Data Area.
Grid	Activate the grid lines in the data window.
Points	Draw the data as discrete points when checked
Shim Units	Cause the Shim Units to appear in the upper right of the Data Display Area.
Auto Scale	Adjust the Display Window parameters to center the displayed data in the NTNMR Data Area. You may toggle the on/off check mark by re-selecting the menu bar. Once the proper display range is determined, the auto scaling may be de-activated to improve display speed. Same as the Toolbar Auto Scale button.
Fit to Window	Perform a single Auto-Scale operation on the data in the current window.
Previous Zoom	Reset the display to the previous Zoom setting
Integrals	Toggle on and off the integral display.
Line Fits	Toggle on and off the Line Fits display.
Peak Picks	Toggle on and off the Peak Picks display.

5.5

5.5 "Option" Menu

Amplitude Adjustment
Baseline Fix
Phase Adjustment
Peak Pick
Integrals
Line Fits
1D Interactive Apod
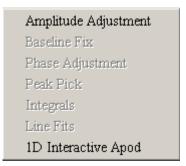
Figure 5.6 Option Pull down menu

Each of the selections in the **Options** menu activates a different processing mode for NTNMR data. See chapter 7 on Options for more information about the available modes.

Menu Items description:

Menu Item	Description
Amplitude Adjustment	Activate/deactivate the amplitude adjust mode. (Default mode)
Baseline Fix	Activate the Baseline menu and allows you to adjust the baseline. (Only in frequency mode)
Phase Adjustment	Activate the Phase area and allows you to perform zero order and first order phase corrections on the spectrum. Auto phase is also available. (Only in frequency mode)
Peak Pick	Activate the Peak Pick menu (Only in frequency mode)
Integrals	Activate the Integrals menu (Only in frequency mode)
Line Fits	Activate the line fitting mode (Only in frequency mode)
1D Interactive Apod	Activate the interactive Apodisation mode (Only is time domain)

5.6

5.6 "Commands" Menu

The **Commands** menu contains a list of data acquisition, data processing and file I/O instructions that can be executed by the NTNMR program. The menu is organized in a hierarchical fashion. Consult chapter 6 for a complete description of each command available.

5.7

5.7 "Script" Menu
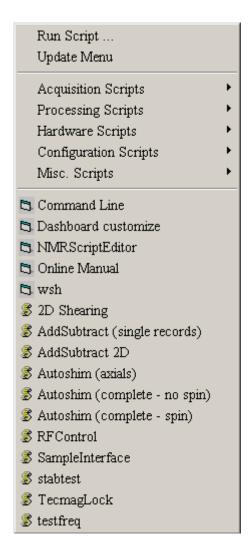
Figure 5.7 Script pull down menu

The **Script Menu** allows scripts to be activated from within NTNMR. See chapter 12 (Automation) for details on creating, modifying and customizing scripts and the script menu.

Menu items description

Menu Item	Description
Run Script	Execute a script by selecting a script to run through a standard File Open dialog.
Update Menu	Update the Script menu if changes are made in the "Scripts" directory while NTNMR is still running.
Acquisition Scripts	List all the scripts in the "NTNMR/Scripts/Acquisition" sub-directory
Processing Scripts	List all the scripts in the "NTNMR/Scripts/Processing" sub-directory
Hardware Scripts	List all the scripts in the "NTNMR/Scripts/Hardware" sub-directory
Configuration Scripts	List all the scripts in the "NTNMR/Scripts/Configuration" sub-directory
Misc. Scripts	List all the scripts in the "NTNMR/Scripts/Misc." sub-directory
Script name list	A list of scripts that are in the "NTNMR/scripts" folder .

5.8

5.8 "Tools" Menu

NMR Data Analysis

Figure 5.8 Tools pull down menu

The **Tool Menu** provides access to advanced data analysis and instrument control features, including some external accessory programs, from within NTNMR.

Menu items description

Menu Item	Description
NMR Data Analysis	The "NMR Data Analysis" option is described in detail in chapter 11

5.9

5.9 "Window" Menu

Figure 5.9 Window pull down menu

The **Window** menu provides items for activating/deactivating various windows as well as a list of currently active windows.

Menu Items descritpion

Menu Item	Description
New Window	Create a new window containing a copy of the current data and parameters.
	(Different than File|New)
Cascade	Re-size and staggers all active windows.
Tile Horizontally	Horizontally tile all open windows into the main window.
Tile Vertically	Vertically tile all open windows into the main window.
Arrange Icons	Organize all minimized windows at the bottom of the main window.
Comment	Open the comment window. Same as the comment button.
Peak Listing	Open a text window that contains the peak listing resulting from a Peak Pick.
Integral Listing	Open a text window that contains the integral listing resulting from an integration operation.
Pulse Program	Activate the Pulse Sequence window. Same as the Pulse Sequence button.
Active Filenames	A listing of all of the open data and sequence windows.

<table>
<tr><td>**5.10**</td></tr>
</table>

5.10 "Help" Menu

Online Guide
Reference Manual
About

Figure 5.10 Help pull down menu

The **help** menu provides a link to the online guide, the reference manual and a window which will give the version information of the software.

Menu items description

Menu Item	Description
Online Guide	The NTNMR Online Guide provides brief topical help for various aspects of NTNMR.
Reference Manual	More detailed information about NTNMR and hardware topics is available in the NTNMR Reference Manual and supporting documentation. The NTNMR reference manual is viewable in Adobe Acrobat format. The reference manual is also fully searchable and is ready for printing.
About	Activate the "About NTNMR" windows shown below. This window gives information about the version of the software you are running and the default locations of specific system files.

Figure 5.11 About NTNMR information window

5.11

5.11 Preferences Dialog

Certain preferences can be set which affect the operation of various aspects of NTNMR. Selecting Edit|Preferences activates a tabbed dialog box that has several settings for the general behavior of NTNMR. The description below outlines the available settings.

5.11.1 Acquisition/Console
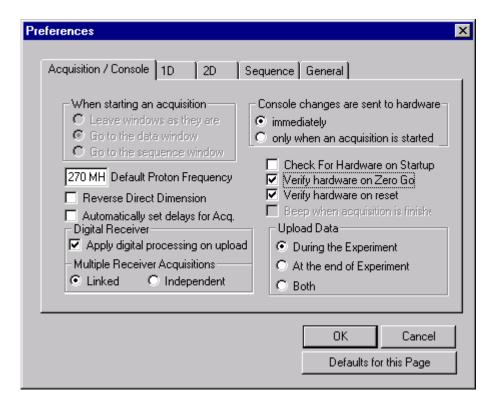
Figure 5.12 Preferences dialog box (acquisition tab)

When starting an acquisition

- Leave windows as they are
- Go to the data window
- Go to the sequence window

These settings determine what the active window is when starting acquisition.

Console changes are sent to hardware:

- Immediately

- Only when an acquisition is started

These settings determine whether or not changes in the console display will be immediate or set only when a new acquisition is started. Items such as frequency change, pre-amplifier selection, etc. will be affected.

Default Proton Frequency

Specify the default proton frequency to be used. This setting has no effect on the hardware. The RF Control application will use this value to calculate the resonance frequencies of various nuclei in its initialization list. The default is also used when a new sequence is created.

Reverse Direct Dimension

Customers using our hardware as an upgrade or an addition to their existing equipment may need to choose the correct direct dimension direction to be able to observe properly oriented spectra.

Automatically set delays for Acq.

Determine whether or not the acquisition time will be set automatically. This can be used to perform an experiment where the dwell time will varry but the user wants to keep the acquisition delay constant.

Check For Hardware on Startup

Determine whether or not to check for hardware when startup. If one of the modules failed to initialize, the user will be informed

Verify hardware on Zero Go

Determine whether or not to verify hardware when hitting Zero Go command. If one of the modules fails between experiments, the user will be informed

Verify hardware on reset

Determine whether or not to verify hardware when hitting reset command. If one of the modules fails to initialize, the user will be informed

Beep when acquisition is finished

Determine whether or not to send a beep when acquisition completes.

Digital Receiver

- Apply digital processing on upload

Tecmag ORION and DSPECT systems contain analog digital converter; Apollo system contains digital receiver.

Unchecking this option with a system containing a digital receiver will require that the data be manually pre-processed. This is done by **Command⎮Data Manipulation⎮Pre-Process Digital Data.** However checking it here will run digital processing automatically.

> *The digital receiver (actually the digital filter stages) produces an unusual looking output waveform. The apparent "beginning" of the data is shifted by several points to the right. At first glance, the extra points before the start of the data seem to be zero. However, closer inspection reveals that they are not exactly zero. In fact, due to the time-symmetric nature of the filters, these points contain information that is necessary to reconstruct the original waveform or more importantly its Fourier transform. Therefore, they cannot be discarded.*
>
> *On the other hand, the time "origin" of the data is clearly shifted to the right. This is a "group delay" phenomenon, similar to that observed with analog filters. Since all of the points are delayed, one must either add some extra points or accept the truncation of the end of the signal.*
>
> *For example, in order to obtain a record that contains 256 dwell time samples, we must sample with (256 + group delay) dwell times. If you then apply Command ⎮Data Manipulation ⎮Pre-Process Digital Data the extra "group delay" points will be incorporated into the signal and you will end up with 256 net points plus the extra points appearing as zeros at end of the record. These zeros at the end can then be discarded by changing "Points 1D" to 256.*

Multiple Receiver Acquisitions

- Linked
- Independent
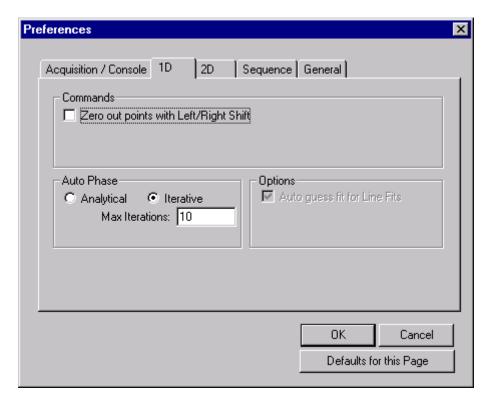
Select the default mode for setting dashboard parameters of each of multiple receivers. When linked, every receiver has same setting; when Independent, every receiver has to be set separately.

Upload Data

- During the Experiment
- At the end of Experiment
- Both

Select the default upload data mode.

5.11.2 1D

Figure 5.13 Preferences dialog box (1D tab)

Commands

- Zero out points with left shift

If selected the left shift and right shift commands will zero the data point(s) that is (are) shifted. If unchecked, the data points will be circularly shifted, appending the shifted point(s) to the end of start of the current data.

Auto Phase
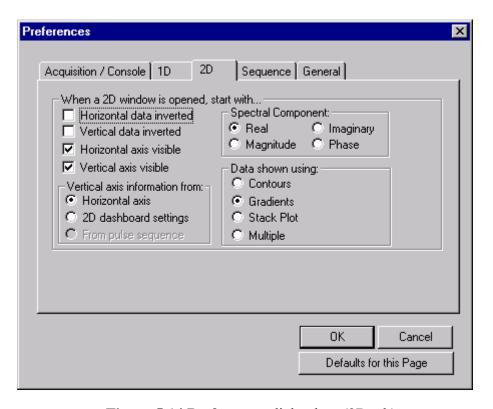
- Analytical
- Iterative
- Max iterations

Selects the default auto phase mode and the default maximum number of iterations for the iterative auto phase mode. See the chapter that discusses the Option modes for more information on auto phasing.

Options

- Auto guess fit for Line Fits

5.11.3 2D

Figure 5.14 Preferences dialog box (2D tab)

When a 2D window is opened, start with…

- Horizontal data inverted
- Vertical data inverted
- Horizontal axis visible
- Vertical axis visible

Vertical axis information from:

- Horizontal Axis
- 2D dashboard settings
- From pulse sequence

If Horizontal Axis is selected, the vertical axis will be an exact copy of the horizontal axis. If 2D dashboard settings is selected, the vertical axis will be calculated from the 2D dashboard settings.

Spectral Component:

- Real
- Imaginary
- Magnitude
- Phase

Selects the data to be shown by default when the 2D data display mode is initially activated.

Data Shown Using:

- Contours
- Gradients
- Stack Plot
- Multiple

Selects the default drawing mode.

5.11.4 Sequence

Not used in the current version of NTNMR.

5.11.5 General
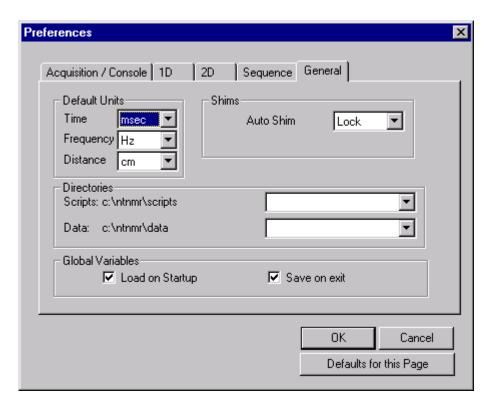
Figure 5.15 Preferences dialog box (General tab)

Default Units

*   Time -sets the default units for time for 1D and 2D displays.

*   Frequency -sets the default units for frequency for 1D and 2D displays.

*   Distance -sets the default units for distance for 1D and 2D displays.

Directories

*   Scripts - indicates the current directory used for scripts.

*   Data - indicates the current default data directory.

Shims
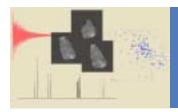
- Auto Shim – set the default auto shim mode.

Global Variables

- Load on Startup
- Save on exit

5.11.6 Defaults for this Page

Reset all the settings back to defaults.

6 - Commands

6.1 Introduction

Most data acquisition and processing commands are listed in the **Commands** menu (Figure 6.1), which is organized in a hierarchical fashion. Other functions such as Set Reference (**View** menu), Auto Phase (**Option** menu), and Integration (**Option** menu) are accessed through other parts of the software.
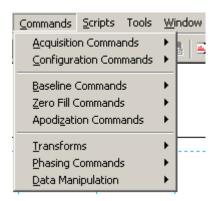
Figure 6.1 Commands menu.

Each of the **Commands** menu items is expanded in the following sections.

6.2 Acquisition Commands
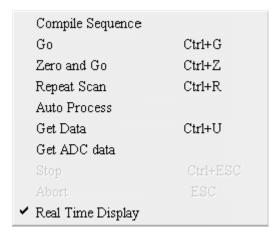
Figure 6.2 Acquisition Commands pull down menu

Compile Sequence	Compile the sequence from the current window.
Go (Ctrl + G)	Start a new acquisition using the sequence and parameters from the current window without zeroing the acquisition memory.
Zero and Go (Ctrl + Z)	Zero acquisition memory and start a new acquisition in the current window.
Repeat Scan (Ctrl + R)	Start the "Repetitive Scan" (RS) mode. In "Repetitive Scan" mode, only the first scan of a pulse sequence is executed, regardless of the Scans 1D, Points 2D, or other dashboard settings. The entire pulse sequence is re-compiled between scans, allowing changes of parameters for tune up, etc.
Auto Process	Enable Auto Processing, which is useful for observing signals in the frequency domain during setup or tune-up procedures, such as shimming or pulse calibration. When Auto Process is on, each FID will be processed using the 1D settings from the NDFT dialog (see the chapter on nD processing for details). A check mark in the menu indicates this option is active. Effective when Real Time Display is set on.
Get Data (Ctrl + U)	Upload the data and parameters from the acquisition memory and replace the data in the front-most window.
Get ADC Data	Upload the current data and parameters from the digitizer (not the averaged data) and replace the data in the front-most window.
Stop (Ctrl + Esc)	Stop the acquisition after finishing the current cycle.
Abort (Esc)	Stop the current acquisition immediately.
Real Time Display	Toggle on and off the real time data display.

6.3 Configuration Commands

Figure 6.3 Configuration Commands pull down menu

Reload Init files … (Ctrl + Shift +R)	Present a standard File Open dialog allowing selection of a 'config.con' file and other initialization files. Opening a new config.con will not affect the currently loaded data windows or sequences. However, any new sequences that are opened will use the information from the new config.con. Reload config.con also executes a Reset Hardware command. See the Pulse Sequence chapter for more information on the config.con file.
Reset Hardware … (Ctrl + Alt +R)	Reset all Tecmag related hardware. (Hardware dependent)
Hardware Status …	Querie all modules installed and return to the operational status. This command writes out a text file to the NTNMR temp directory that contains version information for all installed resources.
Load Grad. Pre-em. …	Present a File Open dialog for opening a set of gradient pre-emphasis parameters.
Save Grad. Pre-em. …	Save the current gradient pre-emphasis parameters from the Dashboard to a file and set them as the default.

6.4 Baseline Commands

Figure 6.4 Baseline Commands pull down menu

Baseline Correction (Ctrl + B)	Remove the DC level of the signal by calculating the mean over the last 12% of the data (Commonly applied before Fourier Transform is performed. Note that the NTNMR Fourier Transform does not apply any DC correction).
Baseline Tilt	Correct for slope and offset of the baseline. (Commonly used before Auto Integrate.)

6.5 Zero Fill Commands

Zero Fill
Echo Zero Fill

Figure 6.5 Zero Fill Commands pull down menu

Zero Fill	Double the number of data points (Points 1D) in memory and fill with zeros.
Echo Zero Fill	Double the number of data points (Points 1D) in memory and fill with zeros, but take into account that the data is in echo format by zero filling around the point specified by the Dashboard parameter "Echo Center".

6.6 Apodization Commands

Exponential Multiplication   Ctrl+E
Gaussian Multiplication
Double Exponential
Sin Bell
Sin Bell Squared
Trapezoid
Traf Window
Blackman-Harris
Echo Exponential Multiplication
Echo Gaussian Multiplication

Figure 6.6 Apodization Commands pull down menu

Exponential Multiplication (Ctrl + E)	Perform an exponential multiplication on the 1D data set with a function defined by: $f(t) = \exp - [\ t * \pi * (LB\ 1D)\ ]$ (LB 1D from Dashboard).
Gaussian Multiplication	Perform a Gaussian multiplication on a 1D data set with a function defined by: $f(t) = \exp - [\ (\ t * \pi * (GB\ 1D)\ ) / (1.6651\ )\ ]$ exp. 2 (GB 1D from Dashboard)
Double Exponential	Perform Gaussian/exponential weighting using the value in DM 1D. (DM 1D from Dashboard)
Sin Bell	Perform multiplication by a sin bell shaped function on a 1D data set with parameters SB Shift, SB Width and SB Skew from the Dashboard.
Sin Bell Squared	Perform multiplication by a sin bell squared shaped function on a 1D data set with parameters SB Shift, SB Width and SB Skew from the Dashboard.
Trapezoid	Perform weighting using 4 data points (TZ 1 1D, TZ 2 1D, TZ 3 1D and TZ 4 1D) as the trapezoid shape delimiters.
Traf Window	Perform Traficante Ziessow weighting using the parameter Traf 1D - see J. Mag. Res., 66, 182-186 (1986).
Blackman-Harris	Perform Blackman Harris weighting using the values of SB Shift and SB Width.
Echo Exponential Multiplication	Perform an echo exponential multiplication on a 1D data set with a function defined by: $f(t) = \exp - [\ t * \pi * (LB\ 1D)\ ]$. The center of the echo, which does not have to be half the number of points, is specified by the Dashboard parameter "Echo Center".
Echo Gaussian Multiplication	Perform an echo Gaussian multiplication on a 1D data set with a function defined by: $f(t) = \exp [- (\ t * \pi * (GB\ 1D)\ ) / (1.6651\ )\ ]$ (GM 1D from Dashboard). The center of the echo, which does not have to be half the number of points, is specified by the Dashboard parameter "Echo Center".

6.7

6.7 Transforms

Fourier Transform	Ctrl+F
Inverse FT	
Real FT	
Echo FT	
NDFT	Alt + `
Hilbert Transform	
2D Back Projection	
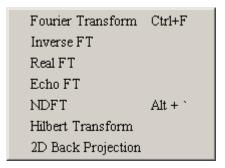
Figure 6.7 Transforms pull down menu

Fourier Transform (Ctrl + F)		Perform a one-dimensional fast Fourier transform on a 1D data set.
Inverse FT		Perform a one-dimensional inverse Fourier transform on a 1D data set.
Real FT		Calculate the one-sided Fourier transform where the imaginary part is set to zero. After the transform of n complex points, to get n/2 complex points back, use the Read 1st Half command. The transform is one-sided, 0 -> +SW, with the first point at zero frequency, so the axis will have to be redefined.
Echo FT		Perform a one dimensional fast Fourier Transform on a 1D data set, assuming that the FID starts in the approximate center of the data at the point specified by the Dashboard Parameter "Echo Center".
NDFT (Alt + `)		Perform a pre-defined set of processing commands on 1D, 2D or 3D data set. The command can be configured to perform baseline correction, zero filling, apodization, transformation and phasing with a single click. To configure each dimension settings of the NDFT command, hold the [Shift] key down and select NDFT. To perform ND FT, just select NDFT after the command has been configured. This command can also be called by clicking on the NDFT toolbar button and configured by holding the [Shift] key and clicking on the NDFT toolbar button.
Hilbert Transform		Perform Hilbert transform on the current data set.
2D Back Projection		Open up 2D Back Projection Settings window. Reconstruct data by projecting 2D records onto specified matrix, assuming uniform angle steps (in degree) are used. Refreshe window with new data.
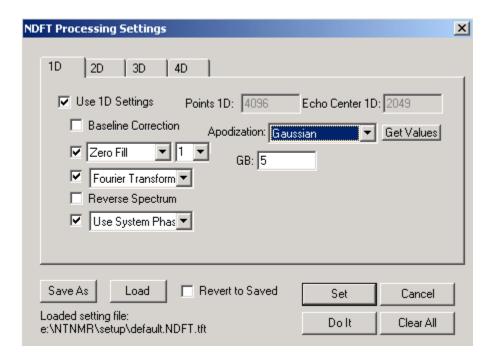
Figure 6.8 NDFT processing settings dialog.

Figure 6.9 2D back projection settings dialog.

6.8 6.8 Phasing Commands
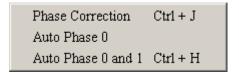
Figure 6.10 Phasing Commands pull down menu

Phase Correction (Ctrl + J)	Apply the zero and first order phase correction in the Dashboard to the current data set.
Auto Phase 0	Use the DISPA method (an analytical method) of auto phase correction to determine and apply the zero order phase correction. (See the chapter on Options for more information on phasing)
Auto Phase 0 and 1 (Ctrl + H)	Use the DISPA method or an iterative method of auto phase correction to determine and apply the zero and first order phase correction. Select [SHIFT] Auto Phase 0 and 1 to select and configure the two auto phase modes. (See the chapter on Options for more information on phasing.)
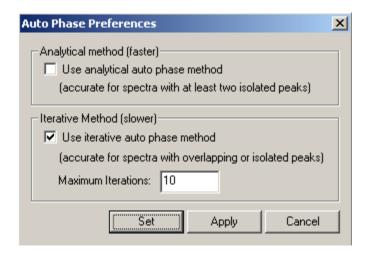
Figure 6.11 Auto Phase Preferences dialog.

6.9

6.9 Data Manipulation

Reverse Spectrum
Read First Half
Read Second Half
Left Shift
Right Shift
Transpose   ▶
Separate Records
Extract Planes
Extract Record
Pre-Process Digital Data
Sort Data Points
Zero Imaginary Data Points
Append ToFile

Figure 6.12 Data Manipulation pull down menu

Reverse Spectrum	Reverse the spectrum in the current data window.
Read First Half	Extract the first half of a data record. The Points 1D parameter on the dashboard is halved.
Read Second Half	Extract the second half of a data record. The Points 1D parameter on the dashboard is halved.
Left Shift	Move the complex data "Shift # Pts." left and fill the extra space with zeros. See the 1D Preferences for the option to circularly shift data points.
Right Shift	Move the complex data "Shift # Pts." right and fill extra space with zeros. See the 1D Preferences for the option to circularly shift data points.
Transpose | Complex	Transpose the X and Y dimensions of a 2D data set.
Transpose | Hyper Complex	Perform a hyper complex transpose of the X and Y dimensions of a 2D data set.
Separate Records	Allow separation or joining of multi-echo data sets. When selected, a dialog prompts the user to select Separate or Join and the number of "echoes" to separate or join into. For example, executing "Separate 1 record into 2 segments" on a 256 x 256 data set would create a 128 x 512 data set where the second record would contain the data from the points 129 - 256 of the first record. Similarly, executing "Join 2 records into 1" on a 256 x 256 data set would create a 512 x 128 data set where the first record would contain the data from the original second record appended to the original first record.

Extract Planes	Extract planes from a 3D data set into separate 2D data sets. A dialog prompts the user to select XY, XZ, or YZ planes. The data files are saved with the names 'basefilename.plane.index_number.tnt' into the directory specified. Use the Browse button to specify the destination directory for the data.
Extract Record	Extract the active record of a nD data set and save it as a data file in the same directory as the current data set. Meanwhile, this file is automatically opened in a new window.
Pre-Process Digital Data	Apply a correction for digital acquisition points at the beginning of a FID. This command should only be used if the system is equipped with a digital receiver and if the "Process Digital Data on Upload" flag is turned off in the preferences dialog.
Sort Data Points	Separate data acquired from multiple receivers.
Zero Imaginary Data Points	Zero imaginary data points of the current record.
Append to File	Used for 2D data manipulation to append the current record onto the data file specified by user.

Figure 6.13 Separate Records dialog.
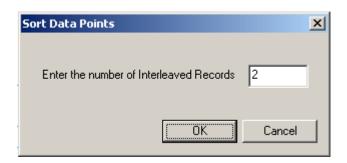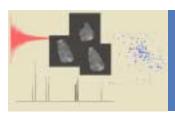
Figure 6.13 Sort Data Points dialog.

7 - Options
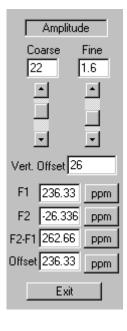
7.1

7.1 Introduction

This chapter covers the various modes that are available from the **Option** menu. The **Option** modes utilize a 'Control Swap Area' that appears to the left of the NMR data in each document window. The Control Swap Area is only displayed if an option is active, and can be completely de-activated by de-selecting the current mode from the **Option**s menu.

7.2

7.2 Amplitude Adjustment

The amplitude adjustment control provides for precise scaling of the 1D data display. The vertical offset of the data can also be set directly. Note that the unit display button activates a pop-up menu that provides for changing the current units displayed. See the preferences section in the Menus chapter for information on setting the default units for time, frequency and distance.

The keyboard and mouse can also be used to scale and zoom on data. Left-click and drag in the 1D display area to select a zoom region; click the selected region to zoom. Using the up and down arrow keys with the [SHIFT] key held down to adjust the offset of the data while the up and down arrow keys without the [SHIFT] will scale the data. The left and right arrow keys move the cursor. The Intellimouse wheel with and without the [CTRL] can also be used to adjust the scale and offset of the data display.

Figure 7.1 Amplitude adjustment control

Amplitude Summary	
Coarse (value and slider)	Number of bits displayed. Adjustable in real time from the slider or by direct input into the text field.
Fine (value and slider)	Multiplying factor of coarse (normal range is 0.5 to 2.0). Adjustable in real time from the slider or by direct input into the text field.
Vertical Offset	Adjust the vertical offset. Enter a value directly or use [SHIFT], up arrow and [SHIFT], down arrow on the keyboard.
F1	Set the left edge of the zoom limit.
F2	Set the right edge of the zoom limit.
F2-F1	Set the width of the displayed frequency interval.
Offset	Set the offset (left/right) of the current zoom.
Units Pop-up	Select the units to display on the axis. (See below)
2D and 3D Record	If a 2D or 3D data set is open, additional controls will appear that allow the navigation through multi-dimensional data sets.

Three axis units, msec, sec and points, are available for time domain. To change the axis units, left-click on the current units to reveal a pop-up menu with the current unit checked. All parameters will be updated for the current selection including the current axis labels displayed in the main data window.

Figure 7.2 Time unit pop up
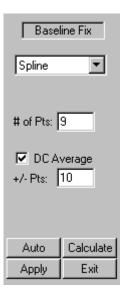
Five axis units are available for frequency domain. To change the axis units from Hz to kHz, MHz, ppm or points, left-click and select the desired unit on the popup menu. All the parameters will be updated for the current selection.

Figure 7.3 Frequency unit pop up

7.3

7.3 Baseline Fix

The Baseline Fix mode (different from Baseline Correction command) is activated by selecting **Baseline Fix** from the **Option** menu. Baseline Fix provides the ability to apply Spline and Polynomial baseline functions to frequency domain data.   Note that Baseline Fix often improves the quality of integration. The table below summarizes the various Baseline Fix functions.

Figure 7.4 Baseline fix control

Baseline Fix Summary	
Spline/Polynomial Selection	Allow selection of either a Spline fit or a least squares fit to an nth order Polynomial.
Order	Enter an integer between 1 and 12 for determining the order of the polynomial fit. *Only visible for Polynomial fits.*
# of Points	Enter an integer between 1 and 50 for the number of points to be used.
DC Average	Toggle on/off the DC averaging function. When on the number of points specified below in '+/-' points will be averaged about each point in the fit to calculate a baseline value.
Auto	Click to automatically calculate point placement for the current data set.
Calculate	Calculate the baseline function. The function will be drawn in the data display area.
Apply	Apply the current Baseline Fix to the active data set.
Exit	Exit Baseline Fix mode.

7.3.1 Working with Baseline Fix

Adding and Removing Points

To manually add baseline points, right click and select Add Point from the right click menu. Right clicking on an existing point will activate the Delete Point option from the menu.

Repositioning Baseline Points
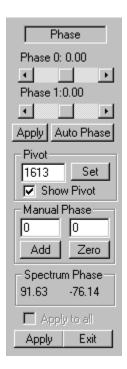
Click and drag on any baseline point to move it to a new position.

Scaling the Display

As in the normal 1D data display the data can be scaled using the up and down arrow keys on the keyboard (as well as the Intellimouse, if equipped). It may be necessary to adjust the scaling and offset of the data to see the calculated function on the display.

7.4 Phase Adjustment

7.4

The Phase Adjustment allows you to perform manual zero and first order phase correction on the spectrum. Selecting **Option|Phase Adjustment** causes the Control Swap Area to change to the Phase Area, as shown in figure7.5. Phase Adjustment can be performed on frequency or time domain data, although only the zero-order correction has any significance in the time domain. Each control item is discussed below.

Figure 7.5 Phase adjustment control

Phase Adjustment Summary	
Phase 0 : Value	Slider to manually adjust the 0 order phase correction. The data display is updated in real time. Changes are not saved until the Apply button is clicked.
Phase 1 : Value	Slider to manually adjust the 1st order phase correction. The data display is updated in real time. Changes are not saved until the Apply button is clicked.
Apply	Apply the current Phase 0 and Phase 1 values set through the manual adjustment sliders and zero the Phase 0 and Phase 1 manual sliders.
Auto Phase	Execute the Auto Phase command. [SHIFT]-click to open the auto phase set up dialog. See below for a detailed description.
Pivot value	Display the current phase pivot value. The phase pivot cursor position is displayed in the data window as a blue cursor. (see Show Pivot below.)
Pivot Set	Set the phase pivot point to the current cursor position.
Show Pivot	Toggle the display of the phase pivot cursor (shown in blue) in the data window.
Manual Phase	Two text entry fields for zero (left) and first (right) order phase parameters to be entered directly.
Add	Apply the current phase values displayed in the Manual Phase text boxes.
Zero	Zero the values in the Manual Phase text boxes. Does not undo previous phasing operations.
Spectrum Phase	Display the phase 0 (left) and phase 1 (right) corrections that have been applied to the current data. Information only.
Apply to all	When checked, the phase corrections applied while in the phase adjustment mode apply to all records in a 2D set. For example, it is useful for applying uniform phase corrections to all records in a 2D experiment or a T1 experiment.
Apply	Apply the current phase corrections to the data, exit the Phase Adjustment mode and update the phase parameters in the Dashboard.
Exit	Exit the Phase Adjustment mode leaving the data and Dashboard unchanged.

The **Phase** Adjustment area displays the total **Spectrum phase** applied at the bottom of the window, which may or may not be equivalent to the **System Phase** (the values used by the **Phase Correction** command in the **Commands** menu). To set the values that will be applied when you select **Phase Correction** from the **Commands** menu, edit the parameters "Sys. Phase 1D" and "Sys. Phase 2D" in the Dashboard.

The slider bars in the **Phase** area may be used to add or subtract zero and first order phase increments to the current phasing of a spectrum. The temporary view phasing display mode in the spectrum window draws the phasing changes

selected by the slide bar controls, but does not apply them to the data unless you hit the **Apply** button. The pivot point used for the first order phasing is also displayed (in blue) in this area.

To manually input data, click on the desired edit box below the words **Manual Phase**. The **Add** button will apply these values to the data immediately without displaying the phasing in the temporary view mode first! **Zero** will reset the Manual input values to zero, but will not undo the changes to the spectrum.

The **Spectrum phase** section will normally display the same values as the values used to correct the spectrum, since the Spectrum phase is updated to reflect the total phase applied to that spectrum. Any additional incremental corrections or Manual input values are added to the **Spectrum phase.** These values are stored in the data file when the phased spectrum is saved. When a previously phased spectrum file is read, the Spectrum phase section is updated. In this case, the Spectrum phase displayed may be different from the System phase. The **OK** button will close the window and change the System phase to the new Spectrum phase values read from the file, and hence will change the values used by the **Phase Correction** command.

Selecting the **Exit** button will UNDO the Spectrum Phase Correction by the values applied from this window and close the window. The **Exit** button does not change the previous System Phase Values. **Note that the value of the zeroth order Spectrum Phase (once OK is clicked) is re-calculated with a pivot point at the center of the spectrum to assure that the Phase Correction command is not dependent on a selection of a pivot point.**

Phasing Help

The two scroll bars, **Phase 0** and **Phase 1**, directly control the zero and first order phase correction parameters according to the formula:

$\emptyset(i) = (\text{Phase } 0) + [(\text{Phase } 1) * (i - i_0) / \text{\#points in spectrum}]$.

The correction is only previewed in the display window until the **Apply** button is hit. Every time the **Apply** button is pressed, the phase correction in the top two fields is applied to the spectrum. The bottom row of numbers (Spectrum Phase) is a record of the cumulative total phase applied to that spectrum.

Phasing in zoom mode is also possible, however first order changes in one part of the spectrum may not adequately represent phase changes across the entire spectrum. The Mini Display Area can be used as a guide, but does not provide adequate resolution to accurately judge phasing. You can return to the full spectrum display by double-clicking in the Mini Display Area, or you can drag
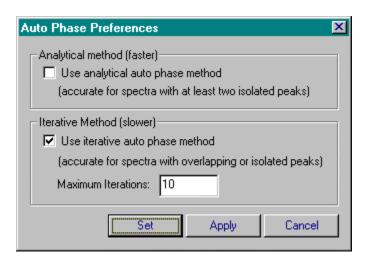
the highlighted zoom area in the mini-display window to move to another zoom region.

A productive approach to phasing is often to correct for Phase 0 errors in a Zoom region near the **Pivot Point** and then to slide the Zoom region to another area of the spectrum far away from the Pivot Point and adjust for Phase 1 error. Iterating via this approach will often yield excellent results.

The **Real**, **Imaginary** and **Magnitude** parts of the spectrum can either be displayed together or separately during phasing by choosing the appropriate items from the **Display** menu.

7.4.1 Auto Phase

The **Auto Phase 0 and 1** command allows you to choose which auto-phasing algorithm is used for the automatic phase correction. Once you choose a method, that method will be used every time **Auto Phase 0 and 1** is chosen until the method is changed or when NTNMR is restarted. The default method when NTNMR is launched is the analytical DISPA method. To change the algorithm used, hold down the [SHIFT] key when you choose **Auto Phase 0 and 1** from the **Commands** menu or [SHIFT]-click on the **Auto Phase** button in the **Phase Adjustment** area. The Auto Phase setup dialog is shown below.

Figure 7.6 Auto phase Preference dialog

7.4.2 Analytical Method

When "Use Analytical Auto Phase Method" is chosen, the DISPA method is used. The DISPA method is described in the Journal of Magnetic Resonance

(JMR 76, 458-478 (1988) by Edward C. Craig and Alan G. Marshall).   This analytical method is reasonably accurate for zero and first order phasing when there are 2 isolated peaks in the spectrum. This method is used for the Auto Phase 0 command and will work with only one peak in the spectrum. This method is the fastest of the two available modes.

7.4.3 Iterative Method

When "Use iterative auto phase method" is chosen, an iterative method of automatic phasing is used, which maximizes the number of points in the baseline and minimizes the width of the peak bases. This algorithm was developed at Abbott Laboratories of Chicago, IL, by Dr. Richard Stephens, and is used in NTNMR with the permission of Abbott Laboratories. This method is good for spectra with overlapping peaks, which do not phase well with the DISPA method. The default number of iterations is 10, which is the smallest number of iterations that yielded reasonably phased spectra in our test data sets. You may change the number of iterations, but any additional iterations will cause this method to take more time. Turn on "ignore auto phase errors" to have NTNMR disregard any errors returned from the auto phase routine. These errors usually occur when the maximum number of iterations has been reached; however, the spectrum is usually phased well enough after 10 iterations to be useful and so turning off the error reporting is usually safe.

7.4.4 Set, Apply and Cancel Buttons

Click "Set" to save the current settings, "Apply" to apply the current settings to the front data window and "Cancel" to leave the settings as they were before the dialog was entered.

7.5 Peak Pick

7.5

Figure 7.7 Peak Pick control
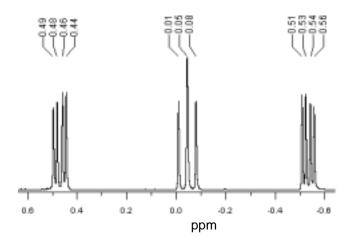
The **Peak Pick** routine is entered by selecting the **Peak Pick** item under the **Option** menu. Choosing the Peak Pick option causes the Control/Swap Area to change from its current selection to Peak Pick, as shown in the figure.

The current Peak Picks, if any are present, can be displayed by selecting **Peak Pick** from the **View** menu without the need for entering the peak Pick Option mode.

Note that in the NTNMR Data Window, Peak Pick are displayed only by label, i.e. a number assigned to each peak found ordered from left to right in the display. When printing, the user has the option to display either the peak pick labels or the chemical shift values directly on each peak.

Figure 7.8 An example of a printed PeakPick display.
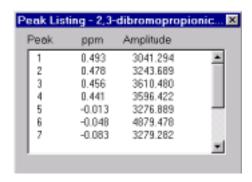
Peak Pick Summary	
Auto Pick: Level	Value above where peaks are detected. NTNMR calculates an initial level threshold.
Auto Pick: Noise	Value which data points must vary by to be considered a peak. NTNMR calculates an initial noise threshold.
Calculate	Calculate the Level and Noise Parameters for the current display area. If an area has been highlighted by a click-and-drag operation, the Level and Noise parameters are calculated for the highlighted region.
Apply	Execute an automatic peak pick using the current Level and Noise parameters.
Remove All	Remove all Peak Picks.
Manual Pick: Add	Add a peak pick at the current cursor position. Manual picks appear in blue. <u>If a region of the spectrum is highlighted, the Add button will execute an Auto Peak Pick for the highlighted region only using the current Level and Noise parameters.</u>
Manual Pick: Remove	Remove the peak pick at the current cursor position regardless of whether it was generated automatically or manually. <u>If a region of the spectrum is highlighted, all picks that fall inside the highlight will be removed.</u>
Label Peaks	Turn on and off peak labeling on the Data Display Area.
Keep Manual Peak on Apply	If checked, all manual picks that have been made will remain after an auto pick has been executed. Otherwise, manual picks will be cleared when an auto pick is executed.
View List	Open the Peak List window. (See below)
Save List	Save the current peak list to a text file.
Exit	Exit the Peak Pick Mode.

7.5.1 General Information

The Peak Pick area provides two edit boxes: Level and Noise. The Level threshold sets a limit on the height of peaks selected. The Noise threshold tests for the difference between potential peaks when moving from point to point in the peak search.

NTNMR will calculate default values for Level and Noise if no values are saved with the data. To set a value for Level or Noise, enter a new value in the appropriate text box. The Level can also be adjusted by positioning the mouse pointer at the top of the level area on the display by dragging up or down.

The Peak Listing window is shown below. A list of currently picked peaks along with the associated chemical shift and amplitude are shown. The peak listing can be saved as a text file by clicking on the Save List button.

Figure 7.9 The peak-listing window.

7.6 Integrals

Choosing the **Integrals** option causes the Control/Swap Area to change from its current selection to **Integrals**, as shown in figure 7.10. The current integrals, if any are present, can be displayed by selecting **Integrals** from the **View** menu without the need for entering the Integration Option mode.

The NTNMR Integration mode allows for a great deal of flexibility in adjusting individual integral parameters. It is essential that the user reviews and understands the various functions of the integration routine for optimal results.

Figure 7.10 Integrals control

Integrals Summary	
Auto Integrate: Threshold	A floating-point number determines if a peak is to be considered for integration. By default, it is 7% of the maximum peak in the spectrum.
Auto Integrate: Noise	A floating-point has value 1.5 times the average of all points that are not above the Peak Threshold. This value, in conjunction with the Shoulder Points, will determine where an integral begins and ends.
Auto Integrate: Shoulder	A long integer value that determines how many points will be averaged as Auto Integrate scans away from a peak.
Auto Integrate: Calculate	Calculate the Threshold, Noise and Shoulder values for an auto integration. Values are calculated for the current data region displayed, or if a region is highlighted for the highlighted region.
Auto Integrate: Auto	Execute an auto integrate command and draw the results on the data display window. Only the data region displayed is integrated. If a region of the spectrum is highlighted, only that region is integrated.
Reset	Reset the active integral to the default values assigned when the integral was originally added.
Scale	Automatically scale the active integral box to the display. Note that this does not effect the integrated value.
Zero All	Zero the slope, offset and curvature values.
Add	Add an integral box over the selected region of the spectrum. If there is no highlighted selection a single integral will be added from the entire spectrum.
Remove	Remove all integrals included in the selected region of the spectrum. If there is no highlighted selection all integrals will be removed from the spectrum.
Labels Pop-up	Determine what type of label will be drawn on each integral. 'No Labels' turns off the integral labels on the display. 'Area Labels' displays the area under each integral. 'Number Labels' displays an identification number for each integral.
Display Multiplier	The Display Multiplier button will cause the multipliers set in the Integral Information window to be shown on the Data Display area.
Use DC Average	Allow a user-selectable number of points around the starting point of the integral to be averaged together and used as a dc offset value. Note: if the DC Average button is off or the number of points to average is zero, the first point of the integral is used as a dc offset.
+/- Points	Number of points to be used in calculating the DC average when DC Average is selected.

Integrals Summary Continued	
Load	Load and integral (*.int) template file from disk.
Save	Save the current integrals as an integral template file (*.int).
View List	Open the integral display list. (See below)
Save List	Save the current integral list as a text file.
Exit	Exit the Integral mode.
Align	Align all of the integrals.

7.6.1 General Information

The **Auto Integrate** command scans the data set and chooses integrals based on characteristics of the data set. Auto Integrate is available from the **Commands** menu.

Integrals are calculated using the following formula:

data_buffer[n] - dc_offset + (curvature * n) + slope
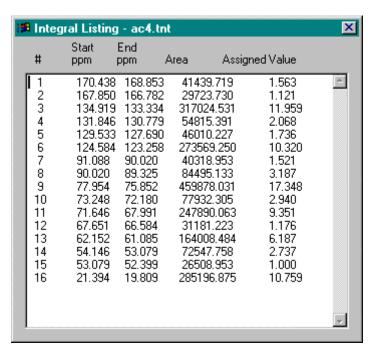
The **Baseline Tilt** command is a command that is used in conjunction with **Auto Integrate**. Choose **Baseline Tilt** from the **Commands** menu either before or after **Auto Integrate** has been applied. This command corrects for the slope and offset of the baseline. Using this command will level off the ends of the integrals created by **Auto Integrate**, since the baseline is corrected. If you have already corrected the baseline using a spline or polynomial fit via **Baseline Fix** from the **Options** menu, do not use this command.

The three characteristics or parameters associated with the **Auto Integrate** are the **Peak Threshold**, **Noise Value and Shoulder Points. The Peak Threshold** is a floating-point number that determines if a peak is to be considered for integration. By default, it is 7% of the maximum peak in the spectrum, and is calculated exactly the same way as in the **Peak Pick** mode.   If a peak is above the threshold, it will be included in an integral. The **Noise Value** has a floating-point value 1.5 times the average of all points that are not above the Peak Threshold. This value in conjunction with the Shoulder Points, will determine where an integral begins and ends. The **Shoulder Points** is a long integer value that determines how many points will be averaged as Auto Integrate scans away from a peak. Once the average of this number of points is below the Noise Value, the integral is terminated.

The **Auto Integrate** parameters are re-calculated each time **Auto Integrate** is chosen from the **Commands** menu. This means that every time **Auto Integrate**

is chosen, the three parameters will be recalculated and then the integrals will be picked.

The Integral List window is shown below. Information about each integral in the current window is displayed and can be saved (**Save** button) as an integral template (*.int) file. Integral template files can be loaded from the **Load** button.

Integral Listing - ac4.tnt

#	Start ppm	End ppm	Area	Assigned Value
1	170.438	168.853	41439.719	1.563
2	167.850	166.782	29723.730	1.121
3	134.919	133.334	317024.531	11.959
4	131.846	130.779	54815.391	2.068
5	129.533	127.690	46010.227	1.736
6	124.584	123.258	273569.250	10.320
7	91.088	90.020	40318.953	1.521
8	90.020	89.325	84495.133	3.187
9	77.954	75.852	459878.031	17.348
10	73.248	72.180	77932.305	2.940
11	71.646	67.991	247890.063	9.351
12	67.651	66.584	31181.223	1.176
13	62.152	61.085	164008.484	6.187
14	54.146	53.079	72547.758	2.737
15	53.079	52.399	26508.953	1.000
16	21.394	19.809	285196.875	10.759

Figure 7.11 The integral list window

7.6.2 Working With Integrals - Manual Integration

Adding Integrals
Add an integral to the spectrum by clicking and dragging to a select region of the spectrum and then either clicking on the **Add** button or right clicking and selecting **Add Integral**.

Removing Integrals
Remove a current integral by clicking and dragging to select the region of the spectrum that contains the integral and either click on the **Remove** button or right-click and select **Del Integral**.

Adjusting Integrals Using the Integral Parameters Dialog
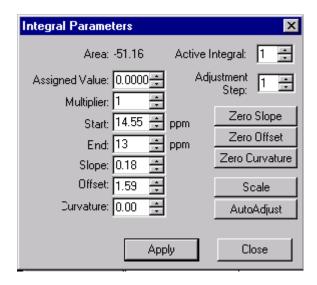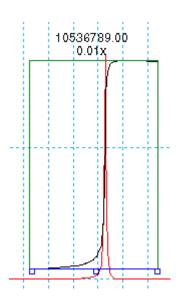
Figure 7.12 Integral Parameters dialog

Integrals can be adjusted by mouse or by parameter. Double-clicking on any integral will open the 'Integral Parameters' dialog that contains adjustments for all parameters for each individual integral in the spectrum. The table below outlines each parameter. See the next section for information on adjusting integrals by mouse.

Integral Parameters	
Area	Display the area of the active integral in absolute units.
Normalized Area	Display the normalized area of the integral.
Active Integral	Select the active integral. All parameters displayed are for the current active integral. The active integral will be highlighted in green.
Adjustment Step	Assign a scaling value (i.e. sensitivity) to all adjustments made in the Integral Parameters dialog as well as by mouse.
Assigned Value	Allow assignment of an assigned value to a particular integral in the spectrum. When a value is assigned to a particular integral all other integrals are given new assigned values. This is used to normalize all integrals in a spectrum.
Multiplier	A scaling value for an integral display. Note that this parameter does not change the integration values.
Start	The start position of the integral is the left-hand side.
End	The end position of the integral is the right-hand side.
Slope	Adjust the integral slope. (See the General Information section for details)
Offset	Adjust the integral offset. (See the General Information section for details)
Curvature	Adjust the integral curvature (see the General Information section for details).
Zero Slope	Zero the active integrals slope.
Zero Offset	Zero the active integrals offset.
Zero Curvature	Zeroes the active integral curvature.
Scale	Automatically scale the active integral.
Auto-Adjust	Auto adjust all parameters for the active integral.
Apply	Apply the changes made in the Integral Parameters dialog.
Close	Close the Integral Parameters dialog.

Adjusting Integrals Using the Mouse

Each individual integral can be adjusted using the mouse in 5 different ways:

- Position the mouse pointer at either the left or right edge to adjust the start or end position of the integral

- Position the mouse pointer at the top of the integral box to adjust the scaling of the integral

- Position the mouse point at the bottom right hand corner to adjust the integral slope

- Position the mouse pointer at the bottom middle adjustment box to adjust the integral curvature

- Position the mouse pointer at the bottom left hand corner to adjust the integral offset

Figure 7.13 An example of the integral display

Note - Holding down the[Shift] key while adjusting an integral by mouse will increase the sensitivity of the adjustment.

7.6.3 Working with Integrals - Automatic Integration
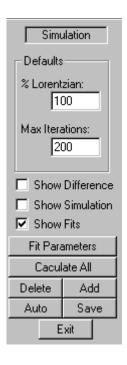
The integration routine allows for automatic integration of an entire spectrum or a selected region. The **Calculate** button will automatically scan the spectrum (or selected region) and adjust the threshold, noise and shoulder parameters. Clicking the Auto button will then apply the integrals. Note that it is sometimes advantageous to allow NTNMR to automatically calculate the threshold, noise and shoulder parameters for selected regions of the spectrum rather than a total calculation.

Using the Auto Integrate function is a quick way to place integrals on the spectrum, and many times will yield acceptable results for routine integration. If the results are not satisfactory, individual integrals can be adjusted, removed or added as discussed in the sections above.

See the General Information section above for detailed information about the Level, Noise and Shoulder parameters.

7.7 Simulation (line fits)

7.7

Line fit mode allows for multiple 'Simulation Boxes' to be added to the active spectrum. The line fit toolbar (shown to the left) controls the general behavior of line fits and the display. Individual line fits parameters are modified by mouse and/or by the Line Fit Parameters dialog discussed below.
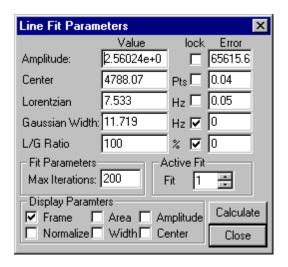
Click and drag to highlight a region of the spectrum where a new line fit is to be added and click the **Add** button to create a new 'Simulation Box'.

The table below outlines the various parameters in the master line fit toolbar.

Figure 7.14 Line fits control

Master Line Fit Parameters	
% Lorentzian	Define the default Lorentzian/Gaussian character of new line fits added. This setting will not override settings in the Line Fit Parameters dialog.
Max Iterations	Set the maximum number of iteration that will be used when a line fit calculation is executed.
Show Difference	Toggle the display of the difference (in blue) between the real points of the active data set and all line fits.
Show Simulation	Toggle the display of the simulated spectrum (in black) resulting from the sum of all line fits.
Show Fits	Toggle the display of the individual line fits.
Fit Parameters	Open the Line Fit Parameters dialog. (See below for more information)
Calculate All	Iterate on all line fits.
Delete	The Delete command will remove any line fits that are completed highlighted by a click and drag operation.
Add	The Add command will add a new line fit simulation box over the currently highlight region of the spectrum.
Auto	Attempt to automatically locate all peaks in the spectrum and add simulation boxes for each peak found.
Save	Save the line fit information for all line fits in the spectrum to a text file.
Exit	Exit the simulation (line fit) mode.

7.7.1 Line Fit Parameters Dialog

The line fit parameters dialog is used to adjust parameters for individual line fits in the spectrum. Each simulation box can be adjusted independently of all others. The table below summarizes all of the parameters in the Line Fit Parameters dialog.

Figure 7.15 Line fit parameters dialog

Line Fit Parameters Dialog	
Amplitude	Report the current amplitude of the active line fit. This value can be edited manually.
Center	Reports the current center point for the active line fit. This value can be edited manually.
Lorentzian	Report the current Lorentzian width for the active line fit. This value can be edited manually.
Gaussian Width	Report the current Gaussian width for the active line fit. This value can be edited manually.
L/G Ratio	Report the current Lorentzian character (in %) for the active line fit. This value can be edited manually.
Max Iterations	Set the maximum number of iterations allowed for a calculate operation on the active line fit.
Active Fit	A spin control that shows the ID number of the active line fit. Click on the up and down arrows to scroll through the parameters for all line fits without exiting the Line Fit Parameters dialog.
Frame	Turn on and off a frame around the active line fit.
Area	Turn on and off the area value display for the active line fit.
Amplitude	Turn on and off the amplitude value display for the active line fit.
Normalize	Automatically normalize all values of the current line fit.
Width	Turn on and off the width value display for the active line fit.
Center	Turn on and off the center value display for the active line fit.
Calculate	Re-calculate the fit for the active line fit.
Close	Close the Line Fit Parameters dialog.
Lock	Each parameter value can be locked by clicking the Lock Check Box next to the value display. A parameter that is locked will not be changed by a re-calculation command.
Error	Report the error estimated for each parameter by the calculation routine.

7.7.2 Working with Line Fits

Line fits are added to the active spectrum by adding individual "Simulation Boxes." Simply click and drag to highlight a region of the spectrum and then click the **Add** button. Once a simulation box has been added to the spectrum, its characteristics can be modified through the Line Fit Parameter dialog as described above or by mouse. In general, the mouse should only be used to make coarse adjustments while the Line Fit Parameter dialog should be used for fine tuning individual line fits.

Manipulating a Simulation Box by Mouse
An individual line fit (Simulation Box) can be modified by the mouse while clicking and dragging at various points on the simulation box.

* Position the mouse pointer on the left or right hand side of the simulation box to expand the overall width of the simulation box

* Position the mouse pointer on the top or bottom of the simulation box to increase or decrease the overall height of the simulation box

* Position the mouse pointer over the small box on the simulated line to increase the line width of the simulation

* Position the mouse anywhere inside the simulation box to move the entire simulation box left or right

Once the line fit has been positioned, use the Line Fit Parameters dialog to perform final adjustments.

General Notes

* It is often helpful to zoom in on a region of the spectrum when manipulating line fits

* Multiple line fits may overlap

* Using the 'Show Difference' and 'Show Simulation' displays helps give a qualitative picture of the quality of individual line fits

7.8	# 7.8 1D Interactive Apodization
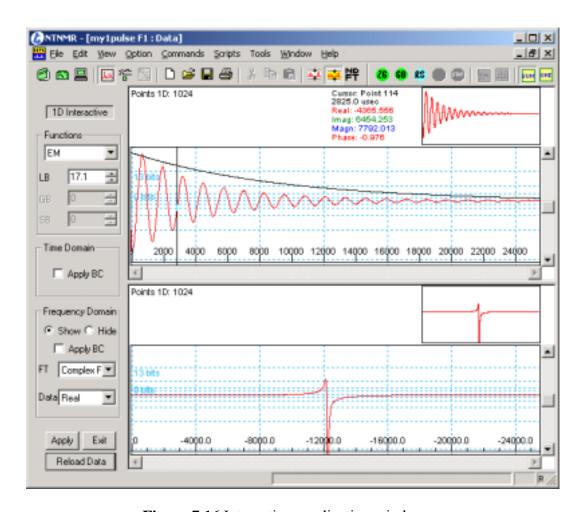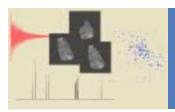
Figure 7.16 Interactive apodization window

Interactive Apodization mode allows the users to perform the available apodization functions like Exponential Multiplication (EM),Gaussian Multiplication (GM), Double EM, Sine Bell, Sine Bell Sqrd, Blackman-Harris, Traf Window, Echo EM and Echo GM, interactively on 1D data sets. It also plots the curve as the selected function is applied to the data set. The data set can be viewed simultaneously in Time and Frequency domain as the selected function is applied to it. The table below describes the various controls available in the 1D Interactive Apodization Dialog Bar.

Interactive Apodization parameters	
Functions	Allow selection of the apodization functions: Exponential Multiplication (EM), Gaussian Multiplication (GM), Double EM, Sine Bell, Sine Bell Sqrd, Blackman-Harris, Traff Window, Echo EM and Echo GM. The appropriate input control LB, GB or SB will be activated based on the selected function. The user can directly enter a numerical value in the edit control or use the spin controls attached to the edit control to increase or decrease the parameter value.
Apply BC	Allow the user to apply the Baseline Correction function to the 1D data set.
Frequency Domain	Allow to simultaneously view the data in the Time and Frequency domains.
Show/Hide	The User can toggle the Frequency domain using the Show/Hide radio buttons. When the Show option is selected the Main NTNMR window is split into two windows and the currently loaded data is displayed in frequency domain in the bottom window.
FT	Allow the user to select the Fourier Transform to be applied to the data set. The available FT options are Real FT and Complex FT.
Data	View the Real or Imaginary data in the frequency domain by selecting the Real or Imaginary option from the selection box.
Apply	The interactive apodization option allows the user to perform various apodization functions and when done with the apodization, the user can use the Apply button to commit the changes to the currently loaded data set in the memory.
Reload Data	Allow the user to discard the changes made to data in the interactive apodizaton mode. Upon selection of this option, a copy of the data set from the memory is made available to the user to perform interactive apodization.
Exit	Exit the Interactive apodization mode.

8 - nD Display & Analysis

8.1 Introduction

8.1

NTNMR has facilities for viewing multi-dimensional data in a variety of ways. Understanding the various view options and which are most appropriate to your data will ensure that you use NTNMR to its fullest. There are four view modes, each mode having several options. Each mode has a main toolbar that controls many of the features. In addition to the main toolbar, auxiliary toolbars may be present. As with all toolbars in NTNMR, the various toolbars specific to multi-dimensional display are dockable and movable and can also be turned off to maximize the display area. Options and controls for the various modes are discussed in detail in the following sections.

8.2 Activating nD Display Mode
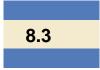
8.2

The multi-dimensional display mode is activated by clicking on the nD display button (Figure 8.1) in the toolbar in an analogous fashion to the way the Pulse Sequence edit window is activated. Also analogous to the pulse sequence editor, each data file can have its own nD display window - the only restriction is available memory. Once activated, an nD display window will appear in the list of open windows in the **Window** menu. Note that the various views associated with a data file such as Pulse Sequence, nD Display, Print Layout, etc. will appear in the Window menu list by the data file name and appended with a number indicating the order in which they were opened. For example, if a data set named ***GRE.tnt*** was opened and then an nD Display window was activated followed by the opening this file's Pulse Sequence window, the Window menu would have the following entries:

- <u>1</u> GRE:Data,
- <u>2</u> GRE.tnt:Data:2048x1024
- <u>3</u> GRE:Sequence

"GRE:Data" would apply to the data file, "GRE.tnt:Data:2048x1024" would apply to the nD display and "GRE:Sequence" would apply to the Pulse Sequence window. The order of the menu listing depends on the order the various windows were opened.

Figure 8.1 nD Display Button

8.3

8.3 nD Display Modes

Several display modes and format options are available for nD display. The most important selections are:

- Data Type (Real, Imaginary, Magnitude, Phase)
- View Mode (Gradient, Multi, Contour, Stack)
- Aspect Ratio (Square, By Data, None)

Some typical examples of data types and the corresponding options are indicated here:

- 2D Imaging data
  - Data type: Magnitude
  - View Mode: Gradient Colors
  - Aspect Ratio: Square

- 3D or Multi-Slice Imaging data: It is usually most useful to first view the data as indicated above for 2D Imaging data, examining individual planes by using the **Plane** spin control and then activating the following settings:

  - Data type: Magnitude
  - View Mode: Multi-Pane
  - Aspect Ratio: Square

- 2D Spectroscopy data:

  - Data type:
    - -Non-phase sensitive experiments: Magnitude
    - -Phase sensitive experiments: Real
  - View Mode: Contours
  - Aspect Ratio: Square or None

- T1/T2/Diffusion data, kinetics data and variable temperature data acquired as pseduo-2D:

  - Data type: Real
  - View Mode: Stack Plot
  - Aspect Ratio: Does not apply

Multi-dimensional display options are not saved with the data file. Clicking on the nD display button will open the nD Display window with current preferences for mode and options. If the mode and options are not the desired ones, they can be changed within the nD display window. However, to save time with screen drawing, options can be set prior to opening the nD Display window by holding down the [SHIFT] key on the keyboard, while clicking on the nD display button. The 2D Setup dialog is shown in Figure 8.2.
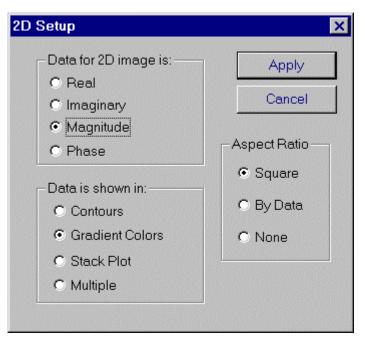
Figure 8.2 2D Setup dialog

The following options can be selected using the 2D Setup dialog:

- Data type - selects the spectral component to be displayed
    - o Real - uses the real data points
    - o Imaginary - uses the imaginary data points
    - o Magnitude - performs a magnitude calculation:
          Sqr. Rt.(R^2 + I^2)
    - o Phase - calculates the phase map:
          arctan (I/R)

- View Type - selects the type of display
    - o Contours - calculates a contour plot of the current data
    - o Gradient Colors - shows the data in continuous color or grayscale mode

        o   Stack Plot - draws a stack plot of the data
        o   Multiple - technically a subset of the Gradient Colors mode. This mode allows up to 25 planes of a 3D or pseudo-3D data set to be displayed simultaneously in a gray scale or continuous color mode.

- Aspect Ratio -sets the display aspect ratio (Note - the data is not altered, only the display)
  - Square - forces the data to be drawn as a square, regardless of the actual number of points in each dimension.
  - By Data - maintains the aspect ratio as dictated by the number of points in the actual data set.
  - None - display is sized to the current display window size without regard to the actual number of points in either dimension.

ND Display options can also be set in the Preference dialog. See chapter 5 for details.

8.4 nD Display Mode Menus

8.4

Only menu items specific to the nD Display mode are covered here. For information on other menu items, consult chapter 5 on NTNMR menus.

File Menu

Save 2D As…	Save the current 2D view as a bitmap (*.bmp) file.
Export…	See chapter 9 on printing and exporting data.
Print Setup	See chapter 9 on printing and exporting data.
Page Setup	See chapter 9 on printing and exporting data.
Add to Print Layout	Add the current 2D view into the print layout of the data file.
Print	See chapter 9 on printing and exporting data.

See chapter 5 on NTNMR menus for other items listed in **File** menu.

Edit Menu

See chapter 5 on NTNMR menus for all items listed in **Edit** menu.

View Menu

Status Bar	Toggle on/off the status bar.
Palette Toolbar	Toggle on/off the color palette.
Gradient Toolbar	Toggle on/off the gradient toolbar.
Contour Toolbar	Toggle on/off the contour toolbar.
Slice Toolbar	Toggle on/off the slice toolbar.
Record Toolbar	Toggle on/off the record toolbar.
Zoom Toolbar	Toggle on/off the zoom bar.
**Cursor	Crosshair**	Activate a full screen crosshair cursor.
**Cursor	Small Crosshair**	Activate a small crosshair cursor.
**Cursor	Vertical Ruler**	Activate a vertical ruler cursor. The size of the ruler cursor is determined with the currently active vertical axis unit.
**Cursor	Horizontal Ruler**	Activate a horizontal ruler cursor. The size of the ruler cursor is determined with the currently active horizontal axis unit.
**Data Type Displayed	Real**	Use the real spectral component for image/contour rendering.
**Data Type Displayed	Imaginary**	Use the imaginary spectral component for image/contour rendering.
**Data Type Displayed	Magnitude**	Calculate the magnitude (Sqr. Rt. (R ^ 2 + I ^ 2))of each complex data point and uses this component for image/contour rendering.
**Data Type Displayed	Phase Map**	Use the arctan (I/R) of each point for image rendering.
**Aspect Ratio	Square**	Force the 2D display to be drawn square, regardless of the number of data points in either dimension.
**Aspect Ratio	By Data**	Use the number of data points in each dimension to define the aspect ratio for display.
**Aspect Ratio	None**	Use the maximum available screen space to display the data without regard to the actual number of points in either dimension.
Set Reference…	Activate the 2D Set Reference dialog. See the section below on setting the 2D reference.
2D Settings…	Activate the 2D Settings dialog. See the section below on 2D Settings.
2D Axis Setup:	Activate the 2D Axis Settings dialog that handles both horizontal and vertical axes.

Re-Calculate <u>L</u>evels on Plane	Toggle on/off a mode when the Plane spin control is used to scroll through the 2D planes in a 3D or pseudo-3D data set, the Color Lookup Table for active plane is automatically re-calculated. If this option is off, the Color Lookup Table for the currently displayed 2D plane will be applied to all subsequently displayed planes in the active data set.

Commands Menu

<u>I</u>ntegrate Selection	Perform a simple integration over the currently selected region. See the section on 2D zooming for details on how to select a region.
Symmetrize…	Update the current view and the toolbars
Sli<u>c</u>e	<u>H</u>orizontal	Activate the horizontal slice display mode. (Slice Toolbar will be activated)
Sli<u>c</u>e	<u>V</u>ertical	Activate the vertical slice display mode. (Slice Toolbar will be activated)
<u>L</u>abel	<u>I</u>nsert	Prompt the user with a text box for adding a text label at the current cursor position.
<u>L</u>abel	<u>E</u>rase	Remove the label (if any) at the current cursor position.
<u>R</u>everse	<u>H</u>orizontal	Reverse the data display in the horizontal direction. (This command is display only, the data is not altered)
<u>R</u>everse	<u>V</u>ertical	Reverse the data display in the vertical direction. (This command is display only, the data is not altered).
Reset <u>V</u>iew	Reset the 2D display area to full display.
<u>P</u>revious Zoom	Set the 2D display area to the previous user defined zoom.

Scripts Menu
See chapter 5 on NTNMR menus for all items listed in **Scripts** menu.

Window Menu
See chapter 5 on NTNMR menus for all items listed in **Window** menu.

Help Menu
See chapter 5 on NTNMR menus for all items listed in **Help** menu.

8.5

8.5 Gradient Color Display Mode

8.5.1 Main Toolbar Controls

Figure 8.3 Main toolbar in Gradient Color display mode

- **Mode**: All nD Display modes have the mode control active. This allows the View Mode to be changed at anytime without the need to exit and re-enter the nD Display. Available options are:

  - Gradient - i.e. continuous colors
  - Contour
  - Stack Plot
  - Multiple

- **Plane**: This spin control allows viewing of all planes in a 3D or pseudo-3D (such as multi-slice data) data set in succession. The number shown corresponds to the currently display plane and can be changed by directly entering a value or by using the up and down arrows.

- **Color**: Allows selection of either Gray Scale or Color display mode. Color mode uses a rainbow style lookup table.

- **Color Levels**: This control allows user selection of the color depth. The settings from 8 bits to 2 bits are available.

- **Start**: Sets the level in the data in arbitrary units at which the first color/gray level will be rendered. (See 'Windowing and Leveling' below).

- **End**: Sets the level in the data in arbitrary units at which the last color/gray level will be rendered. (See 'Windowing and Leveling' below)

- **Spacing**: Sets the spacing between the first and second levels drawn. (See 'Windowing and Leveling' below)

- **Factor**: Sets the multiplier factor used to calculate the spacing for levels 2 through N. (See 'Windowing and Leveling' below)

- **Aspect**: Options for selecting the aspect ratio for drawing the data. 'Square' forces the data to be drawn square regardless of the number of actual points in each dimension. 'By Data' draws the data on the screen in an aspect ratio calculated by the number of points in each dimension. 'None' uses the maximum available space in the nD display window without regard to the actual number of points in the data.

- **Show Grid**: Toggles on and off a grid that is drawn according the to major tick marks chosen for the horizontal and vertical axes.

- **Invert Palette**: Flips the color/grayscale palette that is used to render the data.

- **Smoothing**: Toggles a mode that applies a smoothing algorithm to the data for display purposes only. <u>The raw data is not altered.</u>

8.5.2 Color Lookup Table Toolbar

The Color Lookup Table toolbar (Figure 8.4) allows graphical control over the Start, End and Spacing parameters as well as a visual histogram display of the data.

Histogram: The histogram display gives a visual cue to the relative amount of data at any particular level. The histogram threshold can be set in the 2D Settings dialog as a percentage of the maximum data value.
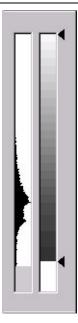
Figure 8.4 Color Lookup Table Toolbar

8.5.3 Windowing and Leveling

The Color Lookup Table has handles that can be manipulated graphically to set the windowing and leveling for a data set. Click, hold and drag the top arrow to move the end level and the bottom arrow to move the start level. Click, hold and drag anywhere on the color bar to move the window over the data. Double-click on the color bar to reset the color distribution to the default values - i.e. start at 1 and end at the maximum data.

The Color Lookup Table can also be manipulated through the Start/End/Spacing and Factor parameters discussed above. Notice that as the graphical tools are used to window and level the data, the Start/End/Spacing and Factor values are adjusted and vice versa.

An extra degree of flexibility is added by using the Factor parameter value. The Factor parameter allows one to apply a non-linear lookup table to the data. A Factor value of '1.00' applies a strictly linear mapping of the color/grayscale levels to the data, meaning that the spacing between each level is identical and is equal to the value in the parameter Spacing. A Factor value of greater than or less than one will apply a non-linear mapping of the color lookup table onto the data according the following:

- Level 1 = Start
- Level 2 = Start + Spacing
- Level N = Start + (Spacing * Factor ^ (N - 2))

Where N = the total number of levels specified

This allows look-up tables to be constructed that emphasize the lower portions of the data by increasing the spacing between levels as the level number increases, or to emphasize the upper portions of the data by decreasing the spacing as the level number increases. This effectively compresses the Color Lookup Table on either end.

8.6

8.6 Multiple Display Mode

The Multiple (or multi-pane) display mode is a subset of the Gradient display mode which allows the simultaneous display of up to 25 planes of a 3D or pseudo-3D data set. This mode is most often used for viewing multiple slices from a multi-slice data set or for viewing planes from a diffusion weighted data set. This mode is restricted to displaying data in the continuous color or gray scale mode.

8.6.1 Main Toolbar Controls

Figure 8.5 Main toolbar in Multiple display mode

Most of the toolbar controls for the Multiple display mode are identical to those in the Gradient display mode. Those controls that are unique to the Multi-Pane display are described below, for others consult the section above on **Gradient Color Display Mode**.

- **Plane**: The plane control in Multi-Image mode moves the focus (indicated by a red outline) to the specified plane. If the layout specified is smaller than the total number of planes, the plane control will "rotate in" the next plane to the display. In other words, if planes one through four of a sixteen plane data set are displayed, when the Plane control counter reaches five, the image display will be re-drawn with images two through five (two at the upper left and five at the lower right).

- **Border**: This spin control sets the number of points to use a border between all panes displayed.

- **Layout**: The Layout combo box allows selection of 2x2, 3x3, 4x4 and 5x5 modes. NTNMR will attempt to fill the available screen space regardless of the total number of planes displayed.

8.6.2 Windowing and Leveling

Windowing and leveling of the data is accomplished as described in the section above on the Gradient Display Mode. The same windowing and leveling parameters are applied to all planes in the data set. If the Mode control is used to exit Multiple display mode, the current windowing and leveling parameters are carried over into the new mode.

8.6.3 Zooming

All zooming functions as discussed in the section on zooming below are available in Multiple display mode. As with windowing and leveling, all planes will have the same zoom parameters applied to them. Therefore, any image displayed can be used to define a zoom region. When the zoom operation is executed all images will update with new boundaries.

8.7

8.7 Contour Display Mode

8.7.1 Main Toolbar Controls

Figure 8.6 Main toolbar in Contour display mode

Most of the toolbar controls for the Contour display mode are identical to those in the Gradient display mode. Those controls that are unique to the Contour display are described below, for others consult the section above on **Gradient Color Display Mode**.

- **Color**: Allows selection of "B/W", "Two Color" or "Ramped Color" display modes. The mode selected determines the colors that will be used to draw the contours. "B/W" draws all contours in black and white, therefore lends no distinction to positive or negative contours in the data. The "Two Color" mode draws all positive contours in blue and all negative colors in red.

"Ramped Color" (the default mode) uses a color ramp to define the color assigned to positive and negative contours. Positive contours are drawn starting with light blue and ending at dark blue so that levels that are closer to "Start" are drawn lighter and the level color darkens as the level increases. For example, more intense data is drawn with a darker shade of blue. Negative contours are drawn starting with yellow and gradually changing to orange and ending in red. For example, the data closest to "Start" are drawn in yellow, closer to "End" are drawn with orange or red. See also the "Invert Palette" selection below.

- **Positive # Levels**: This text box controls the number of positive contours that will be drawn. The check box allows positive contour levels to be turned off.

- **Negative # Levels**: This text box controls the number of negative contours that will be drawn. The check box allows negative contour levels to be turned off.

- **Start**: Sets the level in the data in arbitrary units at which the first contour level will be drawn.

- **End**: Sets the level in the data in arbitrary units at which the last contour level will be drawn.

- **Spacing**: Sets the spacing between the first and second levels drawn.

- **Factor**: The Factor parameter allows one to apply a non-linear lookup table to the data. A Factor value of '1.00' applies a strictly linear mapping of contour levels on the data, meaning that the spacing between each level is identical and is equal to the value in the parameter Spacing. A Factor value of greater than or less than one will apply a non-linear mapping of the color lookup table onto the data according the following:

  - Level 1 = Start
  - Level 2 = Start + Spacing
  - Level N = Start + (Spacing * Factor ^ (N - 2))

  Where N = the total number of levels specified.

- **Aspect**: Options for selecting the aspect ratio for drawing the data. 'Square' forces the data to be drawn square regardless of the number of actual points in each dimension. 'By Data' draws the data on the screen in an aspect ratio calculated by the number of points in each dimension. 'None' uses the maximum available space in the nD display window without regard to the actual number of points in the data.

- **Show Grid**: Toggles on and off a grid that is drawn according the to major tick marks chosen for the horizontal and vertical axes.

- **Invert Palette**: Flips the ramped color palette used to assign colors to contour levels. This selection inverts the "positive" palette and the "negative" palette independently. When selected positive levels are drawn from darker to lighter blue, and negative levels are drawn from red to orange and to yellow. The base color (blue for positive and yellow-orange-red for negative) is not inverted. See the description of the "Color" parameter above for more information.

8.7.2 Color Lookup Table Toolbar

The Color Lookup Table toolbar allows graphical control over the Start, End and Spacing parameters as well as a visual histogram display of the data.

The histogram display gives a visual cue to the relative amount of data at any particular level. The histogram threshold can be set in the 2D Settings dialog as a percentage of the maximum data value.

8.8

8.8 Stack Plot Display Mode

8.8.1 Main Toolbar Controls

Figure 8.7 Main toolbar in Stack Plot display mode

Most of the toolbar controls for the Stack Plot display mode are identical to those in the Gradient display mode. Those controls that are unique to the Stack Plot display are described below, for others consult the section above on **Gradient Color Display Mode**.

- **Whitewash**: When selected, the data is drawn such that the peak in front of any overlapping peaks takes precedent.

- **Scale**: This spin control allows the overall scaling of the stack plot drawn to be adjusted. Values can be directly entered or the up and down arrows can be used for adjustment.

8.8.2 Manipulating a Stack Plot

There are three handles that can be used to manipulate the aspect ratio, position and overall size of stack plots. Click-hold-and-drag on any of the three handles to interactively manipulate the drawing. The speed can be manipulated and can be increased by toggling off the "Whitewash" option.

8.9

8.9 Zooming

The zoom feature is active in Contour, Gradient and Multiple display modes. Zooming for Stack Plots can be done in one of the other modes, prior to selecting the Stack Plot display.

8.9.1 Defining a zoom region:

To define a zoom region, left-click and drag inside the nD display window. Once the zoom region has been defined, release the mouse button. The zoom box has handles in each corner and on all edges for manipulating the size of the box. Click-hold-and-drag on any of the handles to re-size the zoom region. Click-and-hold anywhere inside the defined zoom box to re-position it on the display. Right-click anywhere in the center of the zoom box to execute the zoom.

Note that the aspect ratio of the zoom box is constrained by the currently selected **Aspect.** If the aspect ratio is set to **Force Square** only, square zoom regions are allowed. If the aspect ratio is set to none, any arbitrary rectangle can be used.

8.9.2 Clearing a zoom region:

Left-clicking anywhere outside the defined zoom region to clear the current selection.

8.9.3 Returning to full scale:

Reset View in the **Commands** menu will reset the display to full scale.

8.9.4 Returning to the previous zoom:

Previous Zoom in the **Commands** menu will set the display to the previously defined zoom region.

8.10 The 2D Cursor

8.10

8.10.1 Cursor Types

Four nD cursors are available and can be selected from the View menu under the sub heading Cursor.

Crosshair: Activates a full screen crosshair cursor.

Small Crosshair: Activates a small crosshair cursor.

Vertical Ruler: Activates a vertical ruler cursor. The size of the ruler cursor is determined with the currently active vertical axis unit.

Horizontal Ruler: Activates a horizontal ruler cursor. The size of the ruler cursor is determined with the currently active horizontal axis unit.

8.11 Cursor Information

8.11

NTNMR displays cursor information along the top of the nD display area depending on the active display mode.

Contour Mode: X and Y coordinate information is displayed for the current cursor position in both the current axis units and in points.

The raw data point amplitude is shown for the current cursor position.

Gradient Mode: X and Y coordinate information is displayed for the current cursor position in both the current axis units and in points.

The raw data point amplitude is shown for the current cursor position.

The distance between the last two positions of the cursor is displayed in the units of the currently active axis. Click the cursor to set a point.

8.12 2D Axes

8.12

Horizontal and vertical axes are defined in the 2D Axis Setup dialog (Figure 8.8). The dialog is activated by selecting "Axis Setup…" from the View menu or by double-clicking on either axis (if no axis is visible, double-clicking to in the axis display region will activate the 2D Axis Setup dialog). There are tabs for setting up the horizontal and vertical axes independently in this dialog.
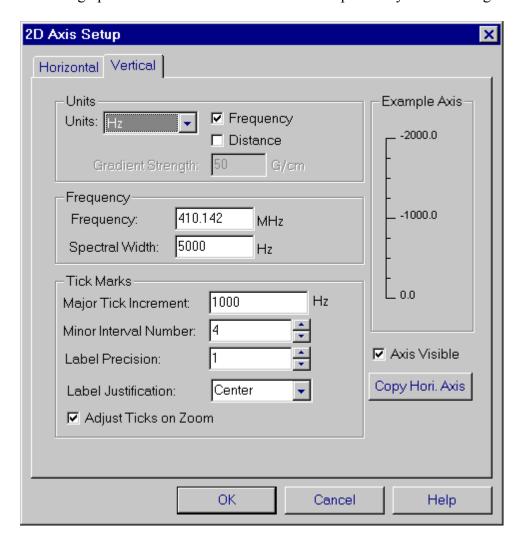
Figure 8.8 2D Axis Setup dialog

- **Units**: Axes can be defined in frequency or distance units. Allowed frequency units are Hz, kHz, MHz, and ppm. Allowed distance units are um, mm, cm, and m. The gradient strength parameter must also be specified for correct calibration of a distance axis.

- **Frequency**: The frequency and spectral width are extracted from the dashboard and are used to calculate the axis. By default the frequency and the SW from the directly detected dimension are inserted into both the horizontal and the vertical axis dialogs. Both parameters can be edited directly by the user.

- **Tick Marks**: The major tick increment defines the base unit for the tick marks that will be labeled. The units defined in "Units" are used. The number of minor intervals, used to calculate the number of minor tick marks, is specified as an integer number. Label precision defines the number of digits after the decimal point that will be included on the major tick labels. The label justification control allows the major tick mark labels to be right, left or center justified. Modifications made to any of the Tick Mark parameters are immediately displayed in the Example Axis.

- **Adjust Ticks on Zoom**: Often when zooming, the tick marks for the defined axis may become too coarse or too fine. This feature activates a mode in NTNMR that will re-calculate the tick marks for the new zoom.

- **Axis Visible**: Toggles on/off the axis in the display.

- **Copy Horizontal Axis / Copy Horizontal Axis**: Copy… allows the user to copy all axis settings from the other dimension.

8.13

8.13 2D Spectral Reference

The reference points for the horizontal and vertical display dimension are set through the 2D Set Reference dialog (Figure 8.9) by right-click on any area of the nD window and select **Set Reference** from pop-up menu, or by selecting **Set Reference** from the View menu. There are tabs for activating the horizontal and the vertical settings independently.
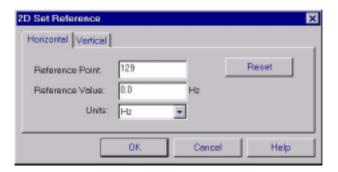
Figure 8.9 2D Set Reference dialog

- **Reference Point**: By default, each time the 2D Set Reference dialog is opened, the current value for the center point in the specified dimension is indicated. This value can be edited so assign a new value to any point in the spectrum.

- **Reference Value**: By default, each time the 2D Set Reference dialog is opened, the current reference value for the center point in the specified dimension is displayed (the center point is indicated as the default entry for **Reference Point**).   The value assigned to the point indicated by **Reference Point** can be entered directly.

- **Units**:   Indicates the currently active axis units. The units can be changed and the units for the reference value will be updated appropriately.

- **Reset**:   Clicking the **Reset** button for either dimension will set the reference value to zero (in the currently selected units) for the center point in the spectrum.

8.14 nD Display Right Click Menu
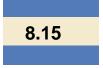
Right-clicking anywhere in the nD display area will open the "right-click" menu (Figure 8.10). Many of the commonly used features of the nD Display mode can be accessed from this menu and is usually the quickest way to work with the display.

Figure 8.10 The "right-click" menu in the nD display window.

- **Set Reference**: Activates the Set Reference dialog. (See section above for details on setting the reference mark)

- **Slice / Horizontal**: Displays a horizontal slice at the current cursor position and activates the slice toolbar. (See Data Slice Tool above)

- **Slice | Vertical**: Displays a horizontal slice at the current cursor position and activates the slice toolbar. (See Data Slice Tool above)

- **Slice | Add Slice:** After Horizontal or Vertical Slice is clicked, click this menu item add current slice to the view. Click and drag the mouse move current slice around and leaves the just added slice in its original position. Multiple slices can be added. The added slices can't be moved or changed.

- **Slice | Delete Slices:** Delete all the added slices, the current active slice won't be deleted.

- **Reverse | Horizontal**: Reverses the orientation of the 2D display about the horizontal direction. (Display only, the data is not altered)

- **Reverse / Vertical**: Reverses the orientation of the 2D display about the vertical direction. (Display only, the data is not altered)

- **Reset View**: Resets the zoom to full

- **Previous Zoom**: Resets the display to the previously chosen zoom region.

- **Data Type**: Selects the spectral component that is used to rendering the display. Choices are Real, Imaginary, Magnitude and Phase.

- **Aspect Ratio**: Allows the aspect ratio of the display to be set to "force square", "by data" and "none."

- **Cursor**: See Cursor Options above.

- **Integrate Selection**: When a zoom box is defined, Integrate Selection performs a simple integration over the selected region.

- **Label**: Insert or erase a text-based label.

8.15

8.15 Data Slice Tool

In Gradient and Contour display modes NTNMR has facilities for viewing 1D slices through the 2D array in the horizontal and/or vertical directions. Selecting Slice | Horizontal or Slice | Vertical from the Commands menu will activate the Slice toolbar (Figure 8.11). The Slice toolbar has two identical groupings of controls, one for vertical display and one for horizontal display.
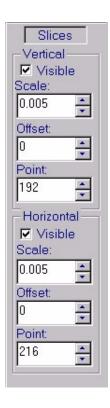
Figure 8.11 Slice toolbar

- **Visible**: Toggles on/off the slice display for the vertical or horizontal direction appropriately.

- **Scale**: The scaling of all slices for each direction can be adjusted independently. Scale values can be entered directly or the up and down arrows can be used to scale the data.

- **Offset**: By default, the active slice is drawn at its corresponding point position in the data display. The offset parameter allows the slice that is drawn to be moved away from its actual point position in the data.

- **Point**: The point parameter specifies the slice of the data that is drawn and is indicated by a slice position line drawn on the display. Entering a value directly or using the up and down arrow keys will cause the corresponding slice to be drawn.

Manipulating slices by Mouse:

Data slices can be moved in either direction by mouse. When the mouse pointer is positioned over the slice position line, a double-headed arrow appears. Click-hold-and-drag to move the slice position line to a new position.

8.16

8.16 2D Preferences

The master NTNMR Preferences dialog is open when Preferences is selected from the Edit menu. It contains a tab for 2D display default settings.

- **Horizontal data inverted**: applies a spectrum reverse to the horizontal axis of the data before display. (Display only, the actual data is not altered)

- **Vertical data inverted**: applies a spectrum reverse to the vertical axis of the data before display. (Display only, the actual data is not altered)

- **Horizontal axis visible / Vertical axis visible**: Determine whether or not horizontal and vertical axes will be displayed as the default behavior. Regardless of this setting, the axes can be activated through the 2D Axis Setup dialog.

- **Spectral component**: Sets NTNMR to display Real, Imaginary, Magnitude or Phase data by default. The data component used to render the data can be changed on the fly in the 2D display mode through the View menu and the right-click menu.

- **Data shown using**: Sets NTNMR to display contours or gradients as the default. The data rendering can be changed on the fly in the 2D display mode.

- **Vertical axis information from:** Vertical axis can be created by copying from Horizontal axis, Dashboard settings or from pulse sequence.

8.17 2D Settings

8.17

The 2D Settings dialog (opened by selecting "2D Settings" from the "View" menu) contains master controls for many of the items that are available on the 2D Display Main Toolbars. Each tab is discussed in turn below.
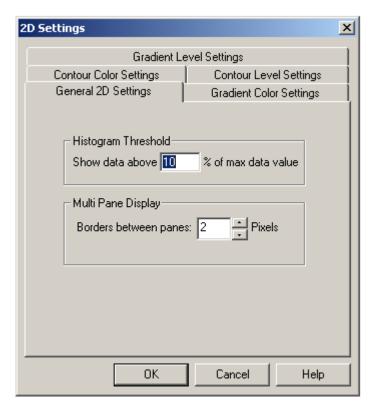
Figure 8.12 2D Settings dialog

8.17.1 Gradient Color Settings Tab

- **Gradient**: Color or Gray Scale can be selected.

- **Gray Scale Intensity**: Slider controls allow the starting and ending gray scale intensity that will be used to map the color palette. (Active for gray scale only)

8.17.2 General 2D Settings tab

- **Histogram Threshold**: Settings are to control the appearance of the histogram in the Color Lookup Table Toolbar. The value entered is the percentage of the maximum data value; all data above the percentage of the maximum value specified will be included in the histogram count.

- **Multi-Pane Display**: When the Multi-Pane display mode (i.e. "Multiple") is active, this parameter sets the number of pixels that will be used as blank space in between panes in the display window.

8.17.3 Contour Level Setting Tab

- **Positive Levels / Negative Levels**: These toggles turn on and off the positive and negative contours respectively.

- **Start**: Sets the level in the data at which the first contour will be drawn.

- **# Positive Levels**: Sets the number of positive levels that will be drawn.

- **# Negative Levels**: Set the number of negative levels that will be drawn.

- **Spacing**: Parameter used to determine the separation of contour levels. See the section above on Contour display mode for details on how contour levels are calculated.

- **Factor**: Parameter used to determine the separation of contour levels drawn. See the section above on Contour display mode for details on how contour levels are calculated.

- **Pen Width**: The pen width (in points) that is used to draw the contours to the screen.

- **Data Range - Data Minimum**: The minimum data amplitude is calculated and displayed. This parameter is informational only and cannot be edited.
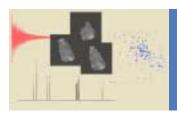
- **Data Range - Data Maximum**: The maximum data amplitude is calculated and displayed. This parameter is informational only and cannot be edited.

8.17.4 Contour Color Settings Tab

- **Contour Palettes**: Choices for Black and White, Two color (red and blue) and Ramped Color are available for contour rendering.

8.17.5 Gradient Level Settings Tab

- **Levels** - Number of Colors: Parameter that determines the number of colors that will be used to render the data display (active for Gradient and Multiple modes only). This number corresponds to the Color Levels parameter in the toolbar.

- **Range - Data Minimum**: The minimum data amplitude is calculated and displayed. This parameter is informational only and cannot be edited.

- **Range - Data Maximum**: The maximum data amplitude is calculated and displayed. This parameter is informational only and cannot be edited.

- **Show Data from**: Sets the start level for the data display.

- **To**: Sets the end level for the data display.

9 – Printing and Exporting Data

9.1 Introduction

9.1

NTNMR supports printing of 1D, multiple 1D and 2D data sets. Printouts can be sent to any available print device or saved to a file using any available printer driver (e.g. postscript or pdf). Note that both 1D and 2D sets can be saved as bitmap or jpeg format using the Export command from the file menu available in the 1D display and 2D display modes but not in print preview mode. Also note that 1D, 2D data sets can be printed in one page. Some other objects can be printed on the same page.

9.2 Basic Printing Operations

9.2

9.2.1 1D Printing

All printing functions are accessed from the Print group in the File menu of the Data Window (Figure 1). Four 1D printing commands are available: Print Setup, Print Layout, Add to Print Layout and Print (Ctrl + P)

```
Print Setup...
Print Layout...
Add to Print Layout
Print...                    Ctrl+P
```

Figure 9.1 Print File menu

Print Setup	Open the setup dialog for the currently active Windows printer. Depending on the printer driver, items such as paper source, page orientation, color management, etc. are set in this dialog.

Print Layout	Activate the NTNMR Print Layout tool. See the Print Layout section for an overview of the various controls for manipulating printouts.
Add to Print Layout	Add the currently displayed data to the Print Layout window, for printing multiple spectra or multiple views of the same spectrum.
Print (Ctrl + P)	The Print command bypasses the Print Layout mode and sends the current data directly to the printer. The default print template (generic.prt) will be used to format the printout (if it exists).

9.2.2 2D Printing

When an nD display window is open and frontmost, the following print selections are available from the File menu: Print Setup…, Page Setup… and Add to Print Layout.

Print…	Open the print dialog for the currently active Windows printer. Depending on the printer driver, items such as paper source, page orientation, color management, etc. are set in this dialog.
Add to Print Layout	Add current nD object to Print Layout. If there is no active layout, the default layout is activated and the nD object is displayed with the 1D record that was opened when the .tnt file was open.

9.3

9.3 Print Layout Tool

The Print Layout Tool allows users to customize most features of their printouts. Print templates that are created in this tool may be saved, and loaded later to apply to other data sets for consistently-formatted printouts.

Select 'Print Layout' from the File menu when in any 1D view to add the current data set to the Print Layout and open the Print Layout mode.

> *Note: This Print Layout can have 1D and 2D data sets simultaneously. This section explains the scenario when 1D data sets are the only kind of objects Print Layout has.*

Shown below is an example of the 1D print layout window.
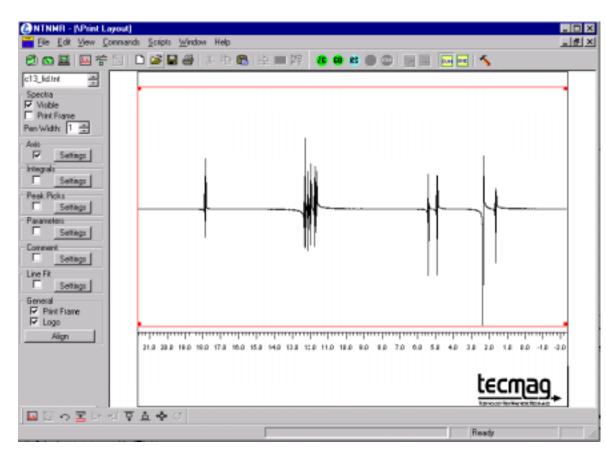
Figure 9.2 1D print layout window example

9.3.1 1D Print Layout

1D Print Preview Toolbar

This toolbar is displayed if there are no 2D data sets on the Print Layout, or the active data set is a 1D data set. If any 2D data set is active, the 2D Print Layout Toolbar is on and 1D Print Layout Toolbar is off.

Data Set Selection Box: The text box at the top of the window shows the filenames for the data sets that have been added to the 1D Print Preview. All other settings except for those in the 'General' section apply only to the data set that is currently active in the Data Set Selection box.

Spectra / Visible: Determines whether or not the selected spectrum will be printed.

Spectra / Print Frame: Determines whether or not NTNMR will draw a frame around the active spectrum.

Spectra / Pen Width: Allows selection of the pen width to be used to draw the active data set.

Axis Checkbox: Toggles on/off the axis for the selected data set.

Axis Settings: Opens the Print Preview Settings dialog to the Axis setup item.

> **Pen:** Determines the pen width that is used to draw the axis for the active data set.
>
> **Font:** Selects the font size to be used for the axis labels for the active data set.

Integrals Checkbox: Toggles on/off the integrals for the selected data set.

Figure 9.3 Print preview tool bar

Integral Settings: Opens the Print Preview Settings dialog to the Integrals setup item.

- **Font:** Sets the font size for displayed integral information.

- **Pen:** Selects the pen width used for the integrals for the active data set.

Other Options: Show Value, Show Frequency Limits, Show Multiplier, Show Slope/Offset/Curvature

Peak Picks Checkbox: Toggles on/off the peak picks (if any) for the selected data set.

Peak Picks Settings: Opens the Print Preview Settings dialog to the Peak Picks setup item.

Font: Selects the font size to be used for the peak picks for the active data set.

Presision: Selects the precision of the picked peak position..

Label Combo Box: Choose whether peaks picks are labeled by number or by chemical shift. The current axis units will be used for label by chemical shift.

Parameters Checkbox: Toggles on/off the parameters display for the selected data set.

Parameters Settings: Opens the Print Preview Settings dialog to the Parameters setup item.

- **Font:** Selects the font size to be used for the parameters for the active data set.

- **Column:** Sets the spacing between each columns of data.

Comment Checkbox: Toggles on/off the comment display for the selected data set.

Comment Settings: Opens the Print Preview Settings dialog to the Comment setup item.

Font: Selects the font size to be used for the comment (if any) for the active data set.

Line Fits Checkbox: Toggles on/off the line fits (if any) display for the selected data set.

Line Fits Settings: Opens the Print Preview Settings dialog to the Line Fits setup item.

Pen: Selects the pen width to be used for the line fits (if any) for the active data set.

General / Print Frame: When selected, a frame will be printed around the entire plot *(this is a global setting for the entire plot)*.

General / Logo: When selected, a logo will be added to the plot *(this is a global setting for the entire plot)*.

Align: Snaps all print layout elements to the invisible grid.

9.3.2 Data Parameters

By default, all parameters from the dashboard are included in the parameter section of the printout. Furthermore, each data set added to the print Layout has its own set of parameters. The specific parameters that are included in the printouts can be customized for each data set in the print layout window.

To customize the parameters that are printed, double-click anywhere on the parameters window and the **Print Parameter** dialog box will appear (figure 9.4), which can be used to select the parameters that will appear on the printout. The current Dashboard parameters appear on the right hand pane and the currently selected parameters to print are in the left-hand pane. Printed parameters can be added, removed or repositioned using the **Add**, **Remove**. **Move up** and **Move Down** buttons.
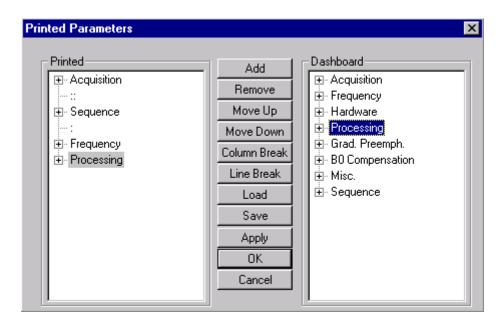
Figure 9.4 Printed Parameters dialog box

Column and line breaks for the printed parameters can also be added.
The Load and Save buttons allow 'Par' files to be read from and saved to disk.
By default, all parameters in the current dashboard are used for all printouts.

9.3.3 2D Print Layout

Select **Add to Print Layout** when in any nD display mode to activate the Print Layout mode (Figure 9.5).
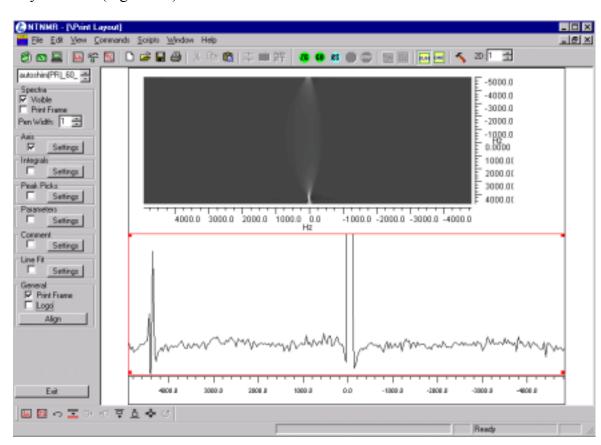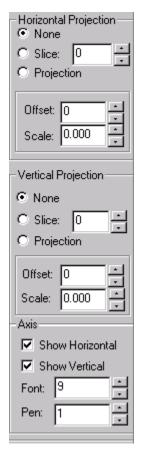
Figure9. 5 nD Print Layout example

9.3.4 Customizing 2D Printouts

This toolbar appear when you select print layout in the 2D display mode.

Horizontal Projection

None: No 1D file will be displayed for the horizontal dimension.

Slice: When a slice selected, corresponding to the number that is displayed to the right, it will be displayed as the horizontal projection.

Projection: will be calculated and displayed by taking the maximum value from all of the data in the horizontal direction.

Offset: Adjusts the offset, in points, for the 1D data displayed from the 2D plot.

Scale: Adjusts the scale factor for the 1D data that is displayed on the horizontal axis.

Vertical Projection

Controls for the vertical 1D data projection are analogous to the horizontal controls described above.

Figure 9.6 2D print layout tool bar.

Axis / Show Horizontal: Toggles on and off the display of the horizontal axis for the 2D plot.

Axis / Show Vertical: Toggles on and off the display of the vertical axis for the 2D plot.

Font / Pen: Sets the font and pen width for 2D Axes

9.3.5 Manipulating Components in the Print Layout

Visible components can be re-positioned and resized by the mouse, no matter if they are 1D data sets or 2D data sets, or some other components. At any time,

only one component on the Print Layout is active. Active components have red outlines and 4 red solid squares on the corners.

Figure 9.7 Print Layout Align Toolbar

1D data set / Visible: Determines whether or not to show 1D data sets for Print Layout. Since there can be multiple 1D data sets, they are handled as a whole. 1D data sets contain not only spectra but also other components that can be controlled on the 1D Toolbar. (Axis, Comments, Peak Picks, Labels, bitmaps…)

2D data set / Visible: Determines whether or not to show 2D data sets. There can also be multiple 2D data sets and they are handled altogether as well.

Undo: Undo last manipulation by user. The manipulation in this context is concerned with the component's position and size only.

Make Next Component Active: Since this Print Layout potentially contains more components than before, it can be pretty crowded. Components may overlap each other. This button enables user to select the particular component to handle. Remember, at any time, there is only one active component and it has a red outline and red squares on its corners. When the shift key is pressed while the user clicks on this button, it will traverse back and forth within the components belonging to one 1D data set.

These 4 buttons push active components to left, right, top, and bottom of the boundary of page respectively.

Dialog button: provides a dialog to determine the active component's position and size numerically.

Rotate Spectrum: Rotates the active spectrum when it is allowed.

Mouse: Visible components can be re-positioned and re-sized by the mouse. To select a visible component, click on the item, when selected, the item will be highlighted with a red outline box and red squares on the 4 corners. Provided that the components are not overlapping with each other, clicking directly on item will perform the selection. If you have many overlapping components, it is

recommended that you press the Tab key or click on the 'Make Next Object Active'.

Re-sizing an Item: Position the mouse cursor at any edge of a highlighted item, depending on the appearance cursor, click and drag to re-size the item. Note that data axes can't be independently re-sized. Data axes are automatically re-sized to match the associated spectrum.

Moving an Item: When the mouse cursor is positioned over an item in the print layout window, the cursor changes to a four-headed arrow. Once the four headed arrow is visible, click and drag to reposition the item.

Positioning a Data Axis: Each individual data set may have its own data axis. Data axes, however, cannot be arbitrarily positioned on the printed page. Left and right movement of a data axis is linked to the position of the actual data. The axis can be repositioned vertically by clicking and dragging up or down in the window.

Popup Menu brought by right click Depending on where the user right clicks, different menus will pop up. For each popup menu, there is always an item called 'Property', which leads user to manipulate that component. Clicking the above toolbar can also offer most of the functionalities the popup menu does. Unique functionalities are as follows:

Insert Object: Insert label or bitmap.

Remove Object: Remove label, bitmap, 1D data set or 2D data set. The primary 1D data set can't be deleted.

Keyboard Specifications:
Tab Key: Pressing the tab key has the same effect as clicking on the 'Make Next Object Active' button. Pressing the tab key and shift key together has the same effect if the user was to press the shift key while clicking on the 'Make Next Object Active' button.

Delete Key: Remove active 1D data set or 2D data set. The primary 1D data set can't be removed.

Left, Right, Up, Down Keys: move active component accordingly. Pressing one of the above keys while pressing the shift key resizes the active component.

9.3.6 Print Templates

As far as one particular data file is concerned, there are three kinds of print templates: Generic Template, Default Template and Other Template.

Default Template: If a particular template is saved with same name as the document, in the same folder, that template is a default print template for that document. It means the next time the user opens Print Layout for this document, the default print template is read. In this case, the actual components are loaded.

Generic Template: it is a global print template that applies to all. Under File menu in Print Layout mode, there is an item called 'Save As Generic Template' that saves the current template as generic, in a particular folder set by Preference. If no default template exists for any particular documents, generic template is read and loaded when Print Layout mode is opened.

Other Template: When Print Layout mode is on, any template (including a default template for another file) can be forced to be read by selecting 'Open Print Template' under File menu. In this case, what are read from 'other template' are just components' sizes and positions.

9.4

9.4 Exporting Data

There are three places that the user can export data to other applications. The first place is from 1D window; the second is from nD window; and the third is from print layout window.

Data formats that are exported from 1D window:

Text file (*.txt):	Create a text file that stores real, imaginary, and magnitude data points if from 1D data set, or others if from nD data set. Also store which NTNMR file (*.tnt) file this text file gets data from, starting and ending points.
MacNMR file (*.mn5)	Create a MacNMR 5.x version data file
Bitmap file (*.bmp)	Create a bitmap file that stores the graphic spectrum, axis, and grids
Jpeg file (*.peg)	Create a jpeg file that stores the graphic spectrum, axis, and grids
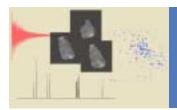
Meta file (*.wmf)	Create a meta file that stores the graphic spectrum, axis, and grids

Data formats that are exported from nD window:

Bitmap file (*.bmp)	Create a bitmap file that stores the graphic stack plot, or gradient plot, or contour plot plus axes
Jpeg file (*.peg)	Create a jpeg file that stores the graphic stack plot, or gradient plot, or contour plot plus axes
Meta file (*.wmf)	Create a meta file that stores the graphic stack plot, or gradient plot, or contour plot plus axes

Data formats that are exported from Print Layout window:

Export As Meta File (*.wmf)	Create a meta file that stores both 1D and nD objects and other objects if there are any according to the current layout of the window,
Copy to Clipboard As Meta File	Copy objects from current layout window to the clipboard. It is ready to be pasted to other applications like MS Word

10 - Pulse Sequences

10.1 Introduction

10.1

This chapter discusses the general features of the NTNMR sequence editor common to all hardware types. This chapter does not cover hardware specific information such as the purpose of a particular line in the sequence.

When the user transfers to the Pulse Sequence window via the **Pulse Sequence** Button on the NTNMR document window, a new window appears for defining a pulse sequence or displaying a pre-defined sequence.

Pulse Sequences in NTNMR are created, edited and displayed in a graphical format analogous to that used to represent sequences in the literature.

Each horizontal line of the pulse sequence corresponds to a function in the spectrometer. The columns of the Pulse Sequence Window represent the events in a sequence, ordered from left to right.

Some features of Pulse Sequence creation and editing in NTNMR are:

- Sequences are always saved with the data file (*.tnt), but can also be saved separately (*.tps) so that they can be used with other data files.

- All sequence tables (phase, delay, shape, etc.) are saved with the sequence and can be created with the built-in text editor of the Edit Table dialog window. This window is easily accessed by right-clicking on any table icon in the sequence or by clicking on the Edit Tables button in the Sequence Toolbar.

- Cut, Copy and Paste commands provide convenient event-level editing within a sequence or between sequences.

- Multi-dimensional tables are segmented into a separate section of the graphical window.

- Tables can be viewed as icons or as table names.

- Unlimited user defined variables can be used. User defined variables appear in the sequence tab of the dashboard.

- Variable values can be computed automatically using Mathematical Expressions.

- Spectrometer configuration is read from a file ("config.con") which specifies what elements can be controlled from the pulse sequence.

10.2 Pulse Sequence Window

10.2

10.2.1 Overview

The Pulse Sequence Window is accessed via the **Pulse Sequence** button on the Main Toolbar in the Main NTNMR Window (or from the Pulse Program selection in the Window Menu). Only one Pulse Sequence Window can be associated with a Document Window (i.e. data file); however, each open data file can have a different sequence window associated with it. The following figure shows a typical Pulse Sequence Window:
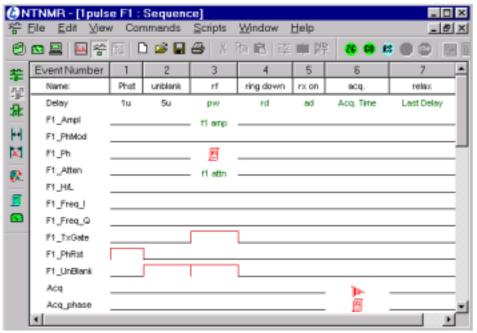
Figure 10.1 NTNMR pulse sequence window

The Pulse Sequence Window has the following components:

- Sequence Toolbar (left side of window).

- Line labels (i.e. "F1_Ampl", "Acq", etc.) associated with various spectrometer functions (installation dependent).

- Event numbers (from left to right across the top of the window).

- Event names (user definable, i.e. "unblank", "relax") are for information only.

- Delay line defining the duration of each event in the sequence.

- Grid (the major part of the window) containing event icons (such as ⬒ , ──── , or ⬚), indicating the action for each line of the sequence during each event. (The grid lines are invisible to improve readability).

10.2.2 Pulse Sequence Toolbar

The Pulse Sequence Toolbar contains several buttons that are used for controlling the appearance of the pulse sequence and for editing the pulse sequence. By default the Sequence Toolbar is "docked" at the left side of the Sequence Window. The Sequence Toolbar can be "undocked" and moved in the same way that other toolbars in NTNMR are manipulated.
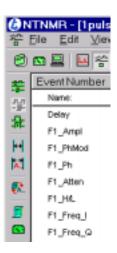
Figure 10.2 the Toolbar docked

Shown below is the Sequence Toolbar "undocked." Note that the toolbar defaults to a horizontal orientation when it is undocked. The toolbar can be positioned anywhere on the desktop and can be docked at any edge of the Sequence Window.
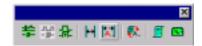
Figure 10.3 The Toolbar undocked

Each of the Sequence Toolbar Buttons is discussed in turn below.

Pulse Sequence Size

Three sequence display sizes are available (small, medium, and large from left to right). Clicking on a sequence size button will cause the sequence to be re-drawn using the new size.

Default Column Width

This button resets the column width to the NTNMR default size. See below for information regarding resizing columns.

Expand Column Width
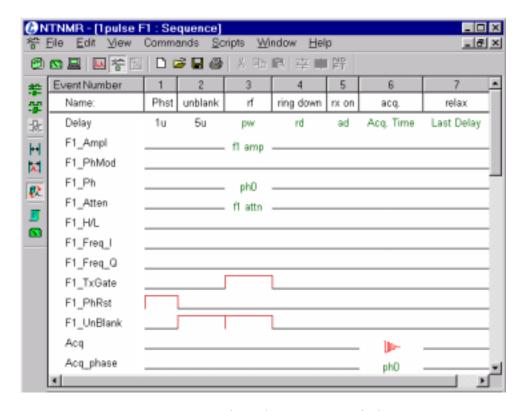
The **Expand Column Width** button automatically resizes all columns so that the complete text name is displayed. In the sequence window example above the columns have been expanded so that the name of Event 6 ("pulse delay") is fully displayed. If the columns are not expanded some user defined event names may be truncated. See the next section for information on resizing individual columns.

Table Name/ Icon Toggle

Tables can be displayed as icons or as text. This button toggles between the two modes. In text mode, the table name is displayed. Note that in either mode, pausing the Mouse over a table for about 1 second will pop up a "Table Tip" containing the table name, a partial list of the table entries (truncated to 16 entries) and the total number of entries in a Tool Tip style box.

The Pulse Sequence Window below shows the Table Name display mode. This represents the same pulse sequence shown in the Table Icon mode in the Pulse Sequence Window above.

Figure 10.4 The pulse sequence window

Edit Tables

The Edit Table window can be opened by clicking the **Edit Tables** button. This is the same as right-clicking on any table in the sequence.

Sequence Bashboard

Clicking on the **sequence dashboard** button opens the Dashboard to the Sequence tab. This allows quick access to all of the sequence variables that have been assigned for editing. Also note that, similar to tables, pausing the Mouse over a variable name in the sequence will pop up a Tool Tip window that displays the variable name and its current value.

10.3 Pulse Sequence Menus

10.3.1 File Menu

The **File** Menu for the Sequence window is identical to the **File** Menu for the Document window with the addition of five Pulse Sequence related items.

New Sequence	Replace the current sequence with a blank sequence.
Open Sequence…	Select and load a new sequence from disk.
Save Sequence	Save the current sequence to disk.
Save Sequence As…	Save the current sequence to disk under a new file name as a *.tps file.
Export Sequence…	Export the current sequence to a text file. This file is simply a list of events and values. *There is no mechanism for importing a text-based sequence.*

The **Print** command in the sequence window will print the current pulse sequence (not the data window) in a graphical format. The number of pages will depend on the length of the sequence.

10.3.2 Edit Menu

Except for "Preferences", the terms in the Edit menu are inactive in the sequence window, unless an event is selected by clicking on the event number at the top of the column.

When an event is selected, the Copy, Cut (and possibly Paste, if a previous Copy or Cut has placed an event on the clipboard) menu options allow clipboard manipulation of the entire event column in the usual Windows manner.

Cut	Remove the currently-selected event column(s) from the sequence and place tem on the clipboard. They can then be pasted elsewhere in the same sequence, or in a different sequence window.
Copy	Copy the currently-selected event column(s) from the sequence and place them on the clipboard. They can then be pasted elsewhere in the same sequence, or in a different sequence window.
Paste	Paste the event column(s) from the clipboard to the event following the selected event column. If no event column is selected, paste at the beginning of the sequence. Pasting is always done by inserting the new event column(s), rather than replacing the selected column.

10.3.3 View Menu

Figure 10.5 Pulse sequence window view menu

Toolbar	Toggle on and off the Sequence Toolbar.
Dashboard	Toggle on and off the Dashboard.
Status Bar	Toggle on and off the Status Bar.
Acquisition Toolbar	Toggle on and off the Acquisition Toolbar.
Console Toolbar	Toggle on and off the Console toolbar (Shims, etc.).
Sequence Tools	
Small, Medium, Large Icons	Select the Sequence Size display. Same as the Pulse Sequence Size buttons on the toolbar.
Details	Activate an alternate text only display of the pulse sequence.
Variable Editor	Toggle on and off the Variable Editor window.

10.3.4 Commands Menu
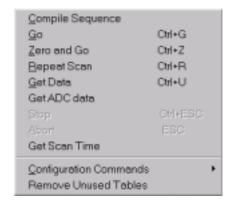
Figure 10.6 The pulse window commands menu

Compile Sequence	Compile the active pulse sequence. *Not required before ZG.
Go	Start an acquisition without overwriting the data in the signal averager memory.
Zero and Go	Start an acquisition, overwriting the data in the signal averager memory.
Repeat Scan	Start the repetitive scan mode. NTNMR runs the current sequence repeatedly until the user issues a stop command.
Get Data	Upload the current data from the signal averager to the host PC RAM and displays the data in the active window.
Get ADC Data	Upload the current ADC data to the host PC RAM and displays the data in the active window.
Stop	Execute a 'soft-stop' at the smallest common multiple of all 1D phase tables. *Same as the yellow stop button on the acquisition toolbar.*
Abort	Execute an immediate stop, regardless of the current phase cycle or acquisition status. *Same as the red stop button on the acquisition toolbar and the ESC key on the keyboard.*
Get Scan Time	Report an *estimate* of the duration of the current experiment.
Configuration Commands	See the Commands chapter.
Remove Unused Tables	Remove unused tables from the current pulse sequence.

10.3.5 Scripts Menu

The Scripts Menu in the Sequence Window allows users to run and write NTNMR Scripts that are saved in the scripts folder. Scripts can be used to manipulate dashboard variables (including sequence variables) and sequence tables. Refer to the chapter on Automation (chapter 12) for details.
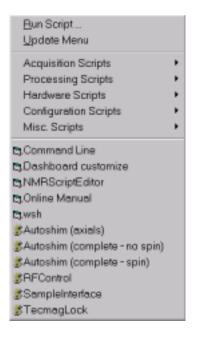
Figure 10.7 The Script pull down menu

Run Script	Select and/or run a script
Update Menu	Updates any changes made in the script directory while NTNMR is still running.
Acquisition Scripts	Select and run the Acquisition Scripts saved at NTNMR/scripts/ Acquisition folder
Processing Scripts	Select and run the Processing Scripts saved at NTNMR/scripts/ Processing folder
Hardware Scripts	Select and run the Hardware Scripts saved at NTNMR/scripts/ hardware folder
Configuration Scripts	Select and run the Configuration Scripts saved at NTNMR/scripts/Configuration folder
Misc. Scripts	Select and run the Miscellaneous. Scripts saved at NTNMR/scripts/ Misc folder
…	Scripts stored at the top level of the NTNMR/scripts/ directory are listed here.

10.3.6 Window Menu

The Window Menu in the Sequence Window has the same functionality as the Window Menu in the Document Window. See the chapter on Document Windows for more information.

10.4

10.4 Defining and Editing Pulse Sequences

10.4.1 Overview

Each position (**Event**) in an NTNMR pulse sequence can be thought of as a "cell" in a spreadsheet representation of a pulse sequence with its own unique X, Y position. An Event's location is determined by its **Event Number** (X position in the "spreadsheet") and its **Line Label,** referred as simply as the **Line** (Y position in the "spreadsheet"). The pulse sequence progresses from left to right, executing all commands in a "column" (i.e. **Event Column**) simultaneously for the duration defined by the **Delay** for that column. Each line represents a specific spectrometer function, such as RF phase, amplitude, acquisition, or transmitter gate.

> *Consult the hardware documentation for your system for a description of the specific sequence lines and their functionality.*

Any particular X, Y position in the sequence is usually referred to simply as an **Event**. This terminology is used throughout this manual. When an entire column in the pulse sequence is referred to, the term **Event Column** is applied.

Any event in the sequence can be modified from the sequence editor by left-clicking on it with the mouse.

An entire pulse sequence can be created in several ways: from scratch, by entering one event at a time; by copying and pasting event columns from an existing sequence; or most commonly, by modifying an existing sequence.

10.4.2 Cut, Copy, and Paste Functions

Sequence **Event Columns** can be manipulated by common Windows-style techniques. When any **Event Number** is left-clicked, the entire column corresponding to that event is highlighted. Highlighted events can be cut (Ctrl-X) or copied (Ctrl-C) using the common Windows keyboard commands or by selecting "Cut" or "Copy" from the Edit menu, or by using the cut/copy/paste buttons on the toolbar. A cut or copied sequence event can be pasted into the

sequence at any position by highlighting an event and selecting "Edit / Paste" (or by typing "Ctrl-V"). Note that Paste inserts the copied event(s) *after* the currently highlighted event, rather than replacing the highlighted event.

Multiple events can be selected by first clicking on an event number and then shift-clicking on another event. All events in a contiguous block will then be selected. The same cut, copy and paste techniques described above apply when dealing with multiple events.

10.4.3 Creating or Modifying an Event

Simply select a spectrometer function by left-clicking on an **Event** in the Pulse Sequence, and choose an item from the pop-up menu list. The menu will change depending on the line you have chosen. Some menu selections will then allow more information to be entered from the keyboard (e.g. the value of a phase or amplitude). To remove an event, choose the 'off' icon ( )from the menu.

10.4.4 Changing the Order of the Lines

The default order of lines that appear in the pulse sequence editor is defined by the configuration file "config.con" (see below). The order can, however, be changed on a per-sequence or per-file basis. This is useful on systems that have many lines, to avoid scrolling the window up and down while editing a sequence. When the line order is changed, the result is cosmetic only and is meant to be an aid in visualizing and editing sequences; the "config.con" file is *not* altered. The new line order is saved with the sequence file.

 An example of moving two lines is shown below:

1. Click on a line label. The line becomes highlighted as shown in the figure. When the mouse is over the highlighted line label, the pointer changes to the "hand" icon.
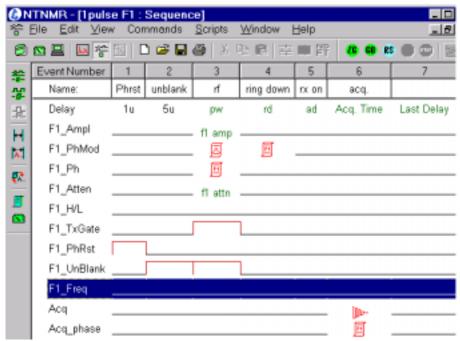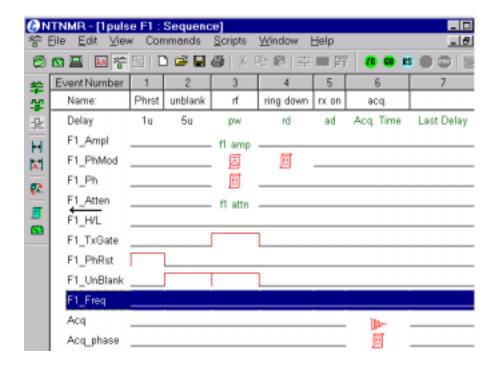
Figure 10.8 A pulse sequence with a highlighted line

2. Click, and hold the left mouse button and drag up or down, away from the highlighted line. The cursor display changes to an arrow (as shown below).

Figure 10.9 Moving a line in the pulse sequence window

3. Move the arrow to the position in the sequence where you want the new line positioned and release. In this example, the line "F1_Freq" was moved from being positioned just above the "Acq" line to just below the "F1_Atten" line.
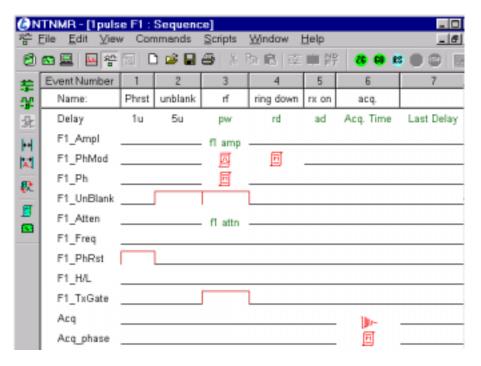
Figure 10.10 Result of moving F1_Freq

10.5 Sequence Elements

This section describes the pop-up menu selections that are available for each type of line in the sequence.

10.5.1 Delay Line

The duration for each event in a pulse sequence is defined in the **Delay Line** that appears as the top line in the sequence window. This line will always be present, independent of the hardware configuration. When a new event is created, either by activating a command for a given position or by using the "Insert" command in NTNMR, a default value of 1u is entered for the new event. The **Delay Line**, similar to the other types of sequence commands, has a unique pop-up menu that it activated when the mouse is single left-clicked on a **Delay** position.

Figure 10.11 Delay line pop up menu.

Edit Delay: The **Edit Delay** selection is context sensitive with two possible options.

1. If an event has a time or a variable name associated with it (as opposed to a table), this selection will allow direct editing of the time value or the variable name for that event.

2. If an event in the **Delay Line** has a table associated with it, the **Edit Delay** selection allows the table name to be edited. <u>In all cases,</u> choosing **Edit…** is <u>the same a double left-clicking on a position in the sequence.</u>

1D Table, 2D Table, 3D Table and 4D Table: These selections activate delay tables. Tables up to 4 dimensions are allowed. 1D tables are created in place and multi-dimension tables are created in a separate location in the sequence window. See the sections on tables and multi-dimensional sequences for more information.

> *All entries in delay tables must have units associated with them (n, u, m, s). Note that "0n", "0u", "0m" and "0s" are invalid entries and <u>will cause fatal errors in sequences</u>.*

10.5.2 Transmitter Type (TX, GR, ET)
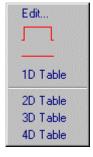
Figure 10.12 Transmitter line pop up menu

The transmitter type pop up menu is generally used for lines that control the transmitter gates (i.e. On/Off) of RF channels on the spectrometer ("Observe" channel, "Decoupler" channel, "F1" channel, "F3" channel, etc.) This pop up may also be used to control the Gain or Attenuation of an RF channel, to activate other spectrometer controls that behave in an "on/off" manner such as the receiver gate, or even to control devices external to the spectrometer such as switches. In some cases, the TX icon may be used to control functions that allow entries of 0 to 100 (floating point), 0-63 in steps of 0.5, etc.

The transmitter type pop-up menu is also used for controlling gradient amplitude lines.

Edit… : The edit selection temporarily converts the display for the current sequence position to a text entry display. The **Edit…** selection has three functions, depending on the current state of the sequence position.

1. If an "**Off**" (see below) gate is assigned to the position being edited, a number can be entered directly. After entering a value and pressing the [ENTER] key, an "**On**" (see below) gate will appear at that position. Note that the number entered must correspond logically to the number of bits that have been assigned for that particular control. For example, if the command that is activated were an RF attenuator that has 8 bit control, the valid entries would be from 0 to 255. Conversely, if the command being edited controls the RF transmitter controlled by 1 bit, then only values of 1 and 0 would be allowed (i.e. "on/off").

2. Choosing **Edit…**: When an "**On**" gate is already active for a position, it will allow editing of the current value for the command.

3. Choosing **Edit…** : When a table is active for that event, the table name will be allowed to be changed. In all cases, choosing **Edit…** is the same a double left-clicking on an **Event** in the sequence.

The "**On**" gate: This command is the equivalent of choosing **Edit…** and assigning a value of 1. This is the common representation of an active event ("on") in a sequence diagram and has the same meaning in NTNMR. Note that for commands that allow greater than one bit control the same "**On**" gate icon is displayed.

The "**Off**" icon: This is the default display for all **Events** in a sequence that are inactive. Selecting the off icon from a pop-up menu of an active position in a sequence will de-activate that **Event** and clear any value that was stored with it.

1D Table, 2D Table, 3D Table and **4D Table**: The 4 table selections will activate a table icon for the current position in the sequence. The 1D table will appear in place as the TX table icon 圓 . For 2D - 4D tables, the icon will appear in a separate section at the bottom of the sequence. See the section on multi-dimensional sequences for detailed information. TX tables can contain lists of any valid data for a particular control. The table editor is discussed in the next section. Regardless of the type of table that is selected, a text entry field will be displayed for assigning a name to the table.

10.5.3 Phase Type (PH)

Figure 10.13 Phase (PH) line pop up menu

The Phase pop-up menu is used for lines in the sequence that control the phase of the various RF channels and of the acquisition.

Edit…: The edit selection temporarily converts the display for the current sequence position to a text entry display. Similar to the **Transmitter** pop-up discussed above, the behavior of the **Edit…** selection is context sensitive. There are two possibilities:

1. If there is no active command for the position, or if there is a fixed phase designated (i.e. +X, +Y, …), a text edit field will appear that allows the user to enter an integer number. This number should be logical with respect to the number of bits that are assigned to the control. A common case would be 2-bit phase control. This typically corresponds to 0, 90, 180, 270 (+X, +Y, -X, -Y) and the control values would be 0, 1, 2 and 3, respectively. Note that the numbers entered <u>are not</u> phase angles, such as 90 or 180, but rather integer numbers that correspond to the number of bits that need to be set for a given phase. While phases of 0, 1, 2 and 3 are common place on many spectrometers, there is also the possibility to have an arbitrary number of bits assigned to the control of a phase shifter. In this case, it is not always

intuitive what integer value corresponds to a given phase. See the section on Tables for information on handling these more complex cases.

2. If a table has previously been assigned to the command that is being edited, a text entry field will appear that allows the name of the table to be edited. <u>In all cases,</u> choosing **Edit…** is the same as double left-clicking on a position in the sequence.
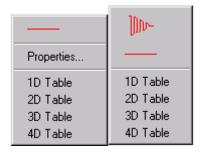
Fixed Phase Selections (+X, +Y, -X, -Y): These selections will assign a fixed phase to a particular event in the sequence. This simply means that no phase cycling will occur for that event in the first dimension. +X equals a value of 0, +Y equals a value of 1, -X equals a value of 2 and -Y equals a value of 3 <u>regardless of how many bits are assigned to that particular line.</u>

The "**Off**" icon: _____ This is the default display for all **Events** in a sequence that are inactive. Selecting the off icon from a the pop-up menu of an active position in a sequence will de-activate that position and clear any value that was stored with that **Event**.

1D Table, 2D Table, 3D Table and **4D Table**: The 4 table selections will activate a table icon for the current position in the sequence. Phase tables are used to program phase cycling schemes for pulse sequences. The 1D table will appear in place as the PH table icon ▦ . For 2D - 4D tables, the icon will appear in a separate section at the bottom of the sequence. See the section on multi-dimensional sequences for detailed information. PH tables can contain lists of any valid data based on the number of bits assigned. The table editor is discussed in the next section. Regardless of the type of table that is selected, a text entry field will be displayed for assigning a name to the table.

10.5.4 Acquisition Type (ACQ)

Figure 10.14 Acquisition line pop up menu

Acquisition icon: The Acquisition icon (represented by an FID symbol) tells the system to begin acquiring data at that event in the sequence. More details are found below in the section on Acquisitions.

The "**Off**" icon: ———— This is the default display for all positions in a sequence that are inactive. Selecting the "off" icon from the pop-up menu of an active position in a sequence will de-activate that position and clear any value that was stored with that position.

1D Table, 2D Table, 3D Table and **4D Table**: The 4 table selections will activate a table icon for the current position in the sequence. The 1D table will appear in place as the SW table icon 🔴 . For 2D - 4D tables, the icon will appear in a separate section at the bottom of the sequence. See the section on multi-dimensional sequences for detailed information. SW tables contain a list of spectral widths. The table editor is discussed in the next section. Regardless of the type of table that is selected, a text entry field will be displayed for assigning a name to the table.

If an acquisition icon is present, left-clicking it will reveal the second popup-menu shown above, with the **"Off"** option discussed above, as well as the **"Properties"** option. Selecting "Properties" will open the Acquisition Properties window. Right-clicking the icon will go directly to the properties window. For more details, refer to the Acquisition section below.
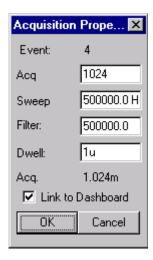
Figure 10.15 Acquisition pop up menu

10.5.6 Shape Type (SH, PS, RM, AT, P2, R2, A2)
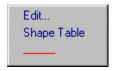
Fig. 10.16 Shape table pop up menu

Edit…: The **Edit** selection temporarily converts the display for the selected **Event** to a text entry display enabling an existing shape table to be re-named. In all cases, choosing **Edit…** is the same a double left-clicking on a position in the sequence.

Shape Table: The Shape Table selection activates a shape table and assigns the icon to that event. Shape tables are primarily for use when generating waveforms for field gradients or RF amplitudes and phases. See the section on tables for more information. Generally, the **Delay** for the event that contains a Shape Table dictates the length of the event. The shape will be automatically scaled to the duration ("Delay") of the event where it appears. So, if the Shape Table has 512 entries and the **Delay** for that event is set to 500 msec, then the table will be clocked out at a rate of (500/512) msec per point. (But see the table editor discussion as well as the alternate "continuation" mode, below)

The "**Off**" icon: This is the default display for all positions in a sequence that are inactive. Selecting the off icon from a the pop-up menu of an active position in a sequence will de-activate that position and clear any value that was stored with that position.

When using the shape icon to specify some waveform types (depending on the hardware), the "**continuation**" mode is also active. As stated above, all shapes are played out during the event in which they are specified with a table time per point defined by the duration of the event divided by the number of entries in the table. However, once a shape has been defined the continuation icon can be placed in subsequent contiguous events to force the waveform to be played out over multiple events. When selected, the icon will appear in the sequence. In this case, the time per point for the waveform will be determined by the sum of the delays for the waveform event, plus all continuation events, divided by the number of points in the table.

10.5.7 Loop Type

> ***Important note regarding loops****: The number of loops that can be executed in a sequence is unlimited. However, the number of loops that can be nested is limited to 4.*

Figure 10.17 Loop table pop up menu

Edit…: The **Edit** selection temporarily converts the display for the current sequence position to a text entry display. The behavior of the **Edit…** selection is context sensitive with several possibilities.

1. If there is no active loop or loop table for the position, the text entry will be used as the number of times the loop is to be executed. After a value has been entered and the [ENTER] key has been pressed, the ↴ (**Loop Start**) icon will appear.

2. If a Start Loop command has been assigned to the current position, **Edit…** will allow the user to enter the number of times that the loop should be executed.

3. If a Loop Table has previously been assigned to the command that is being edited, a text entry field will appear that allows the name of the table to be edited.

4. If a **Loop End** command has previously been assigned to a position, **Edit…**is selected and a value greater than 1 is entered, the command will be converted to a **Loop Start** instruction. In all cases, choosing **Edit…** is the same a double left-clicking on an **Event** in the sequence.

Loop Start: This selection defines the event as the start of a loop. By default, a value of 1 is entered for the number of times to execute the loop. Once a Loop Start instruction has been entered in the sequence, select **Edit…** or double left-

click on the Loop Start instruction to change the value for the number of times to execute the loop.

The "**Off**" icon: _____ This is the default display for all **Events** in a sequence that are inactive. Selecting the "off" icon from the pop up menu of an active position in a sequence will de-activate that position and clear any value that was stored with that position.

Loop End: This defines the end of a Loop. No data value is needed. Loops are executed exactly the same way that parentheses are in mathematics. Thus, the inner most loop will be completed first and the outer most loop will be executed last.

1D Table, 2D Table, 3D Table and 4D Table: There are options for loop tables for each dimension up to 4. Loop tables can be used *only* in place of the **Loop Start** instruction and require a **Loop End** instruction. See the section on tables for information on defining and editing tables.

10.5.8 Asynchronous Sequence

Some function lines in the sequence editor allow another sequence to be loaded (as a sequence file, "filename.tps"). This nested sequence is called an **asynchronous sequence**, as its timing is asynchronous, or independent, of the timing of the main sequence.

This feature allows, for example, complex phase and amplitude modulation schemes on one transmitter channel (decoupler) to run continuously across several events of the main sequence, while the other channels execute the main sequence.
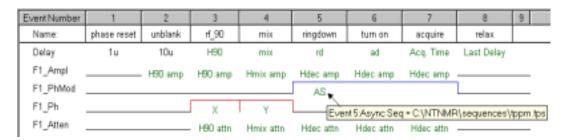
Multiple lines can be grouped and controlled together through a single asynchronous sequence. All lines on the same physical event module constitute a group, e.g., the lines for amplitude control, labeled "Fn_Ampl", "Fn_Attn"; frequency control, "Fn_Freq" (Asynchronous function is not available for G1 frequency control boards); RF gate, "Fn_Gate", (with n as the channel number) within the same channel are grouped in the same asynchronous group. The phase line, "Fn_PhMod", is grouped independently. However, a sequence that uses both amplitude and phase will automatically load both groups.

To activate an asynchronous sequence: left-click on the initial event during which the asynchronous sequence should run. A pop-up menu appears:

Figure 10.18 Asynchronous initial event pop up menu

Select the option "Add Asynchronous Start" and then select the appropriate subsequence file, (e.g. tppm.tps) in the File Open dialog box. To define the end of the asynchronous sequence, left-click the cell where the asynchronous sequence shall stop and select "Add Asynchronous Stop" in the pop-up menu. In the following example, a Two Pulse Phase Modulation (TPPM) proton decoupling (in F1 channel) scheme during data acquisition is implemented into a cross-polarization (CP) main sequence through the asynchronous function. The asynchronous sequence, "tppm.tps" is located in the folder "C:\NTNMR\sequences". It starts with Event 5 and ends after finishing Event 7.

Figure 10.19 Example of the tip displayed when asynchronous sequence is activated

Once the asynchronous sequence is activated, the hardware will implement the asynchronous sequence and ignore the parameters of the main sequence. One activated line in an asynchronous group is sufficient to force the remaining lines to take the provided parameters in the asynchronous sequence. If the subsequence is shorter than the duration of the asynchronous sequence in the main sequence, the asynchronous sequence is repeated as many times as necessary.

The asynchronous sequence is created in the same way as any other sequence file. The figure shows the TPPM asynchronous sequence in the sequence editor containing two events:

F1_PhMod = 12.5° and 8.3 µs duration in Event 1,

F1_PhMod = -12.5° and 8.3 µs duration in Event 2.

The sequence is saved as "tppm.tps", after setting any required parameter values such as delays or phases.

Event Number	1	2
Name:		
Delay	8.3u	8.3u
F1_Ampl		
F1_PhMod	12.5	-12.5
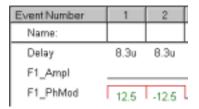
Figure 10.20 An example of a asynchronous sequence

10.6 Tables

10.6

10.6.1 Using the Table Edit Window

Common Features

The Edit Table dialog window is accessed by right-clicking on any table icon in the Pulse Sequence Window or by clicking on the Edit Tables button on the Sequence Toolbar. An example of the Table Edit window is shown below with a phase table active.
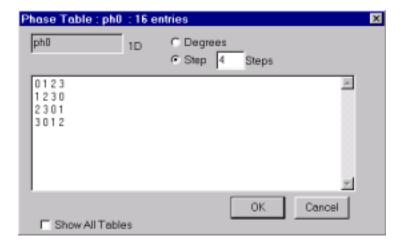
Figure 10.21 The edit table dialog window

A single edit table window is used to edit all types of tables: Phase, Delay, Shape, Gradient, Transmitter and Loop. The type of table is detected upon creation by NTNMR through the module type code for the pulses sequence line. When a table is created, the user may assign it any name. By default, phase tables are named as ph0, ph1, ph2… phN, in the order they were created. Similarly, Delay tables are given the "d" prefix, shape tables are give the "shape" prefix, loop tables are give the "lp" prefix, and transmitter tables are give the "TX" prefix. Again, these prefixes are not mandatory, although they do provide a reasonable scheme for naming tables in a sequence. Also note that, by default, all 2D, 3D and 4D table names are appended with ":*dimension_number* " (i.e. "ph0:2").

Clicking on the **Show All Tables** button will expand the table edit window to display a list of all tables for the current sequence (**List of All Tables** display). Note that the table name and contents are shown in the **List of All Tables** display. However, tables with greater than 16 entries are truncated. The **Show One Table** button toggles the display back to the default.
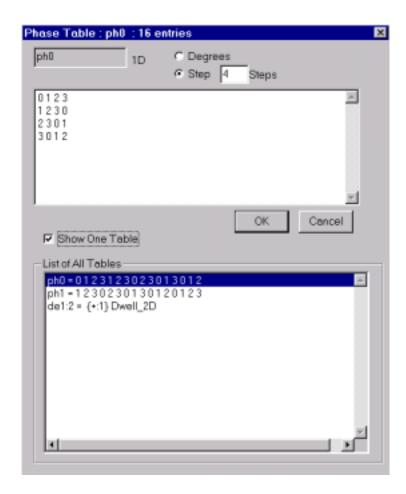
Figure 10.22 The edit table dialog window in expanded mode

When the Table Edit window is in the **Show All** mode, as in the example above, a left-click on any table in the **List of All Tables** window will update the **Edit** window to allow the table selected to be viewed in full and/or edited.

The Title Bar of the **Edit Tables Window** shows the type, name and number of entries in the currently selected table in the format *table_type* : *table_name* : *number_of_entries*.

The table name is displayed in the upper left corner of the window and the dimension of the selected table is displayed to the right of the name.

The **OK** button saves all changes that have been made and exits the **Edit Tables** window.

The **Cancel** button exits the **Edit Tables** window, leaving all tables unchanged.

Editing Tables

The Edit window allows normal text entry similar to a text editor. The mouse can be used to position the cursor and to select text. Windows standard cut, copy and paste operations are available by the "Ctrl + X", "Ctrl + C", and "Ctrl + V" keystrokes, respectively.

Table items *must* be de-limited by either a space ([SPACE]) or a carriage return ([ENTER]). The space and the carriage return are interpreted in exactly the same way when the sequence is compiled. In the example above, the table is formatted using a carriage return after every fourth entry making the table easier to read. The example below shows the same table entered with spaces in between all entries. The two tables are functionally identical. No other characters can be used to delimit entries in a table.
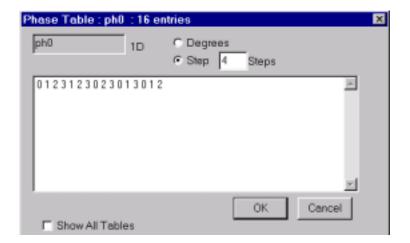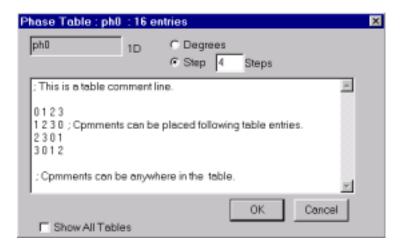
Figure 10.23 The edit table dialog window

Tables can also contain comments. The ";" character (semi-colon) is used to indicate a comment. All text *following* a semi-colon *on a line* is ignored by the compiler. The example below shows the use of the comment character in a table.

Figure 10.24 An example of the use of comments in the edit able dialog window

Renaming Tables
Tables can be renamed by double-clicking on the table name and entering the new name, or by selecting Edit from the Event pop-up menu and typing the new name.   If a table with the new name exists, the existing table will be used for that event. If a table with the new name does not exist, a new blank table will be created.

Removing Tables
Tables can be turned off in the sequence (as can any other active Event) by selecting the "off icon" from the pop-up menu. Tables that have been removed or renamed will remain available in the Table Edit dialog and will still appear in the List of All Tables. This allows the table to be re-used or to be re-activated later if needed. Table can be permanently removed from the List of All Tables by selecting the table in the list and pressing the [DELETE] key.

Selecting "Remove unused tables" from the Sequence Window Commands menu will permanently remove any tables that are not in use in the active sequence.

Importing Tables
NTNMR does not currently have facilities for importing tables directly from a file. However, tables can be copied from a text editor and pasted into the Table Edit window. Tables can also be loaded by an external program via OLE Automation (see Chapter 10). The **Table Wizard**, described below, provides a particularly convenient way to do this.

10.6.2 General Table Information

Table Size
There is no *software* limit on the number of tables of any type or on the number of entries in any given table. There are, however, hardware limits on the length of tables. Please consult the "Hardware Topics: Pulse Programming" guide for more information (see hardware manual).

Multiple Instances
Tables can be re-used an unlimited number of times in the sequence. If, for example, a phase table is created and named "ph1", subsequent events can also use the table "ph1". Double-click on the new (or existing) Event, select Edit from the Event pop-up menu, or select a table of the correct dimension from the Event pop-up and enter the name "ph1".

Table Formats & Units
As mentioned earlier, table entries must be de-limited by a space or a carriage return. Also, a ";" character (semi-colon) can be used to comment on lines in tables.

Time units *must* be included with each entry in Delay tables. The following symbols are valid: n = nanoseconds, u = microseconds, m = milliseconds, and s = seconds.
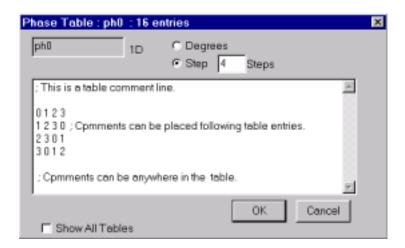
10.6.3 Table Types

Depending on the type of table that is selected, the Table Edit window will display different features. Each type of table is discussed below.

Phase Tables

The Phase Table edit display (shown below) contains a Text Edit window where the table is created/edited as well as the "**Step**" and "**Degrees**" options. The Text Edit window has all of the general features discussed above.

Step mode: The entries in a phase table are *unit-less* numbers that represent a phase (see below for "Degree" mode).

Figure 10.25 the edit table dialog window in degree mode

For the most common case where the phases 0, 90, 180 and 270 are available, the numbers 0, 1, 2 and 3 will most often correspond to these values. This is common notation for phase tables. For cases where a greater degree of resolution is available, and the hardware has been configured appropriately, numbers greater than 3 may be used. For example, if 3 bits of phase resolution is available then number from 0 to 7 are valid.

The **Step Per Cycle** field allows simplified table creation when greater than 2 bits (i.e. 0, 90, 180, 270) control is available. The formula (360/**Steps per Cycle**) is applied to the entire table. So, if the value of **Steps per Cycle** are set to 8, the maximum resolution would be 45 degrees and if **Steps per Cycle** are set to 4, the maximum resolution would be 90 degrees. This allows simplified table creation in cases where, for example, a 90-degree resolution is required, regardless of the number of real bits of resolution available. Note that this value applies on a *per table* basis, not to an entire sequence. The following examples illustrate this:

Example 1
> Hardware phase resolution = 12 bits
> Table Steps per Cycle = 4
> Phase table = 0 1 2 3

In this case, the values of 0, 1, 2, and 3 would correspond to 0, 90, 180 and 270 degrees, respectively.

Example 2
> Hardware phase resolution = 12 bits
> Table Steps per Cycle = 8
> Phase table = 0 2 4 6

In this case, values 0, 2, 4 and 6 corresponds to 0, 90, 180 and 270 degrees, respectively.

Example 3

> Hardware phase resolution = 12 bits
> Table Steps per Cycle = 8
> Phase table = 0 1 2 3 4 5 6 7

The actual phases generated by the hardware are:

> 0 = 0
> 1 = 45
> 2 = 90
> 3 = 135
> 4 = 180
> 5 = 225
> 6 = 270
> 7 = 315

Degree mode: When the degree mode option is selected, phase tables can be entered in units of degrees from 0 to 360. NTNMR will attempt to choose the closest possible phase value that the hardware is capable of producing.

> *NTNMR does not have built in functionality for converting phase values in degrees to the proper logic output for PTS phase synthesizers with the phase rotation option. NTNMR can be used to control the phase rotation feature of the PTS. However, values that are sent to the PTS must be sent as straight 12 bit (or however many bits are being used) number to the PTS.*

Transmitter Tables

Tables can be used in any dimension on transmitter (TX) type events to specify complex transmitter behavior. Common uses for this type of table are for alternately turning on and off a transmitter pulse in a sequence, or if the TX line in the sequence is controlling an attenuator to vary the amplitude of an RF pulse with respect to any dimension. Sample tables are shown below.

The following example shows a 1D TX table that controls the Observe RF pulse in a sequence. In this example, the RF pulse for this event is off in the first scan, on in the second, off in the third, etc.
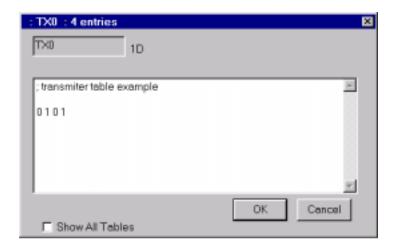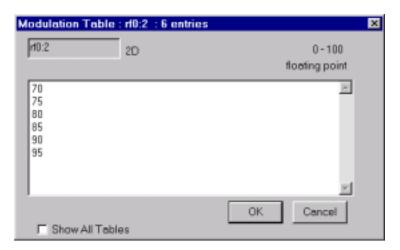
Figure 10.26 An example of the use of tables to control the transmitter gating

This example shows the use of a 2D transmitter table that is controlling the transmitter gain. In this case, each 2D scan would use the next transmitter gain on the list. A common application for this type of table would be a pulse sequence for determining the 90-degree pulse power for a given width.

Figure 10.27 An example of the use of the dialog table to control the transmitter amplitude

Delay Tables

> *All entries in Delay Tables must have units associated with them (n, u, m, s). Note that "0n", "0u", "0m" and "0s" are invalid entries and will cause fatal errors in sequences.*

Two types of Delay Tables are available: manual and auto. The manual table is most commonly used for pseudo-2D type experiments such as T1 relaxation measurements. In this case, a list of delays is explicitly created, along with units, in the Edit window. An example is shown below.
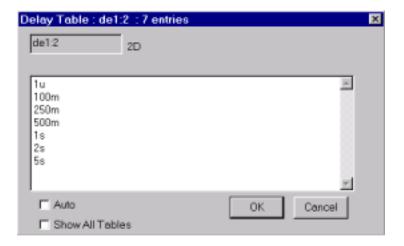
Figure 10.28 An example of delay table

When the **Auto** box is checked, the Edit window display is changed to allow automatic incrementation of a delay table with respect to the dimension of the table. The most common use for a table of this type of for multi-dimensional dwell tables (i.e. t1, t2, etc.).

When in **Auto** mode, the value for the first increment, the first 2D or t1 point in the case of a 2D table, is given by the delay entered for the 1D event in the sequence. Each subsequent 't1' point is incremented by the increment value. The increment value can be specified in one of three ways. First, the increment value can be entered directly into the **Increment Value field**. For the case of the table shown below, the Increment Value has been set to 200u. If the 1D Event Delay time corresponds with this 2D table, set to 1u, then time for that event for the first 2D point would be 1u, for the second it would be 201u, for the third 401u, and so on.
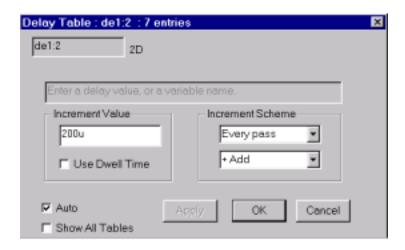
Figure 10.29 Automatically incrementing delay table.

Instead of entering a specific time in the **Increment Value** field, a variable name can be supplied. In this case, (see example below where the variable "TimeInc" is used for the Increment Value) the increment value will be specified by a user defined table variable in the Dashboard. See the section below on Sequence Variables for more information on creating and editing variables.
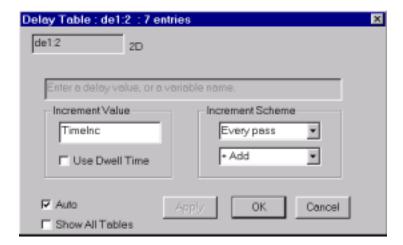
Figure 10.30 Automatically incrementing delay table using a variable

Finally, checking the **Use Dwell Time** option will force the table to use the Dashboard Parameter Dwell_2D for 2D tables, Dwell_3D for 3D tables and Dwell_4D for 4D tables.
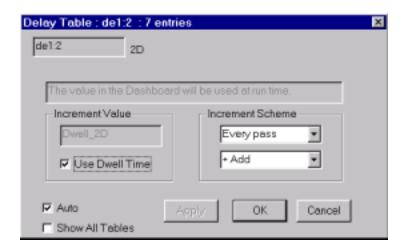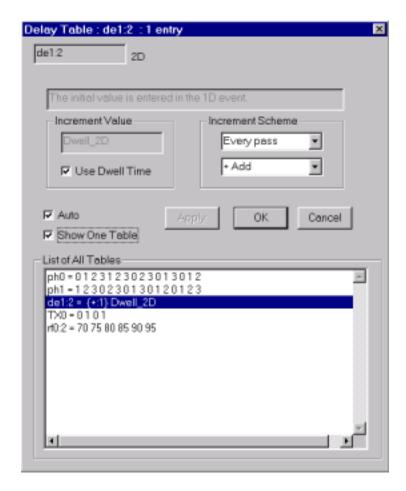
Figure 10.31 Automatically incrementing delay table using the dwell time

Also available in the Auto mode is the ability to define when the table will be incremented. The **Increment Scheme** provides controls for this purpose. The top selection, by default set to **Every Pass**, has the additional choice **Every 2 Passes**. The choice determines when the table will be incremented. If **Every Pass** is chosen, the table will be auto-incremented after *every* 2D point. If **Every 2 Passes** is chosen, the table will be auto-incremented on *every other* 2D point. This is particularly useful in setting up phase sensitive experiment using the States-TPPI method.

The second field, which appears as + **Add** by default, is not currently used.
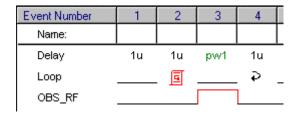
When the Auto selection is chosen for a table the following notation is used in the **List of All Tables** to represent the table: *table_name* = {+:1} *increment _variable_name* for tables that are incremented with every 2D point and *table_name* = {+:0,1} *increment _variable_name* for tables that are incremented with every other 2D point. The notation for tables in Auto mode is shown in the example below.

Figure 10.32 Example of the table dialog box showing all the tables in the sequence

Loop Tables
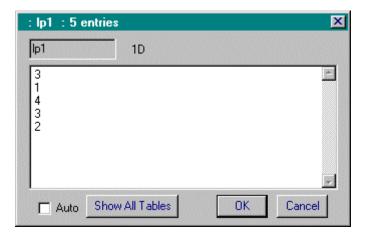
The Start Loop instruction can be replaced with a Loop Table of any dimension. Loop tables allow the number of times that a loop will be executed within a given scan to be set to any arbitrary number with respect to any dimension. An example of the appearance of a Loop table in a sequence is displayed below.

Figure 10.33 Example of the use of a loop in a sequence

In this example, a 1D Loop table has been used as the start loop instruction in Event 2. When sequence execution reaches event 4, the pulse programmer will return to event 2 and repeat the number of times that is indicated in the Loop table for the current scan.

The Table Edit window for Loop Tables is very similar to the Delay Table edit window and has two modes. The "manual" mode allows tables to be created with user defined lengths and with arbitrary order. If the loop table, shown below, was used in conjunction with the sequence example above, during the first 1D scan, the loop from event 2 to 4 would be repeated 3 times. During the second 1D scan, the loop would be repeated 1 time, during the third 1D scan 4 times, etc.

Figure 10.34 Loop table dialog box.

If the **Auto** box is checked, options analogous to those for Delay Tables become available. This mode allows a Loop Table to be auto-incremented according to the scan counter for the dimension of the table. The **Increment Value** field will be the number that is added to the loop counter for each incrementation. The **Increment Scheme** controls allow the table to be incremented after **Every Pass** or after **Every 2 Passes**. The **+Add** field is currently inactive.
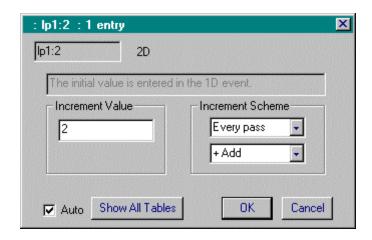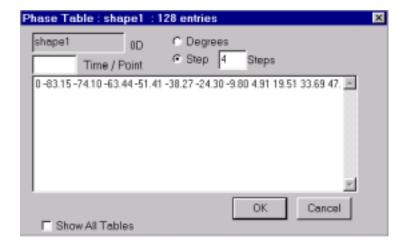
Figure 10.34 Loop table dialog box with **Auto** checked

½D Tables

½D tables are only used to increment a value within in a loop.   These tables are available for use in the gradient amplitude, rf amplitude, rf attenuation, and rf phase sequence lines. When you create a ½D table within a loop it should be followed by an "increment table", also within the same loop. Each time the sequence loops around the next value in the ½D table will be used. Each ½D table on the same sequence line requires there to be an "increment table" following it before another ½D table can be placed on that same line.

Shape Table

A Shape Table is also called "0D" (zero-dimension) table. Only amplitudes (rf or gradient) and phases can have shape tables. The following figure shows a phase shape table as an example.

Figure 10.36 Shape table dialog box.

The hardware sequentially executes all entries of a Shape Table. The Shape Table pop-up windows are analogous to pop-up windows of 1D or 2D … tables. For example, in a phase shape table window, the choices "Degrees" and "Step" are available in a table of any dimension. On the other hand, the option "Time/point" is available only in shape tables, which allows to set dwell time for the table samples. If "Time/point" is empty, the dwell time is given as: (duration of the event)/(number of entries) as discussed above. If a value is assigned to "Time/point", the dwell time is set to the assigned value, and the table repeats as necessary to fill the event durations (including any continuation events). Note that the duration of each entry may be subject to hardware limitations (typically 100ns minimum). (Note the similarity to an 'Asynchronous' sequence). This feature is particularly useful in conjunction with the 'continue table' icon, allowing, for example a decoupling sequence to be carried across several events without resorting to the more complex asynchronous sequence method.

DMOD Tables

A special table type is available for clocking out data tables one entry at a time. Contiguous pulse programmer lines assigned in the 'config.con' file (see the section "Pulse Sequence Configuration File") using the icon library "LD" are classified as "DMOD" (Dedicated MODulator) lines. DMOD is only a description of the line functionality. The actual name assigned to the line in the pulse sequence can be any text string (as "MOD_1" in the example below).

Consider the follow entry in the 'config.con' file:

MOD_1   ES1 11 12 1 0 0 LD 1 0 ;

Lines 0 through 11 (i.e. LP11, LP10, LP9 … LP0) of the module ES1 are defined as DMOD lines.

Any DMOD line in a pulse sequence is controlled, first loading a table and then by activating an "On" gate (same icon as a TX type) to clock the first entry. Each subsequent instance of the "On" gate in the DMOD line will clock the next value in the table until the end of scan, at which time the table will be reset to the first entry.

A DMOD table is loaded by selecting the **"Stepped Table"** choice from the pop-up menu on the DMOD line. The most common application of the DMOD line would involve loading a table in an early event in the sequence, and then toggling the 'On' gate in later events, each time sending the data for the next table entry.

Note that the data output of any lines defined as DMOD lines is *not* latched. However, the data from the DMOD line will be active for the all contiguous "On" events and during the event immediately following the last contiguous event.

DMOD Sequence Example

In the example sequence shown below, a "Stepped Table" was added in Event 2 of the "MOD" line. For the case of this example, assume that the "MOD" line has 2 bits assigned to it in the 'config.con' file.

Consider the following table entries for the Stepped Table in Event 2:

    0 1 2 3 0 1 2 3

The first table entry is clocked out in Event 4 and stays active during Event 5 for a total duration of 15u. In Event 7, the second table value 1, is clocked out and remains active for Event 8. The third table entry, 2, is active for Events 10 and 11. When the scan counter is incremented, after the completion of the last Event (Event 14), the table pointer is reset to the beginning and the first entry is again active for Events 4 and 5.
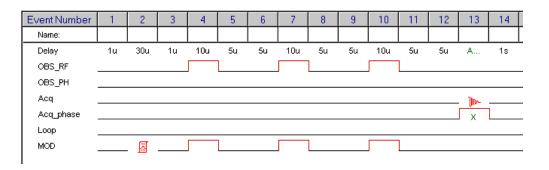
Figure 10.37 DMOD sequence example

10.6.4 Table Wizard
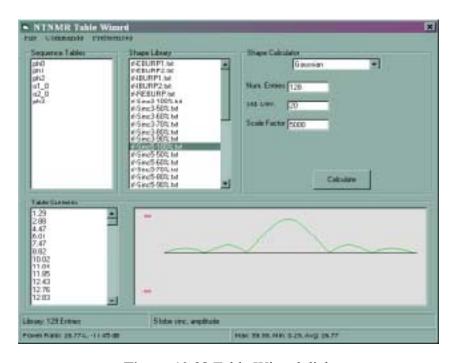
Figure 10.38 Table Wizard dialog

The NTNMR Table Wizard is used for creating new custom shape tables, and also gives access to the shape table library.

Sequence tables. Shows the list of sequence tables associated with open data file.

Shape Library. Shows the list of existing shape tables in the library.
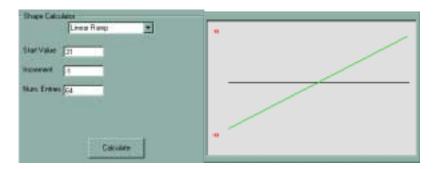
Shape calculator. There are 5 shape types available from the shape calculator: Linear Ramp, Ramp Up/Hold/Down, Sinc, Sine, Gaussian. Each shape form has different parameters that can be varied to get a desired shape table.

Linear ramp.

- **Start Value :** First entry in the shape table
- **Increment:** Step size for calculation of the next table entry
- **Num. Entries:** Total number of entries in the table
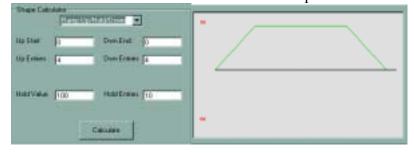
Figure 10.39 The shape calculator

Figure 10.40 Shape calculator with the display on (linear ramp example)

Ramp Up/Hold/Down.

- **Up Start:** Starting value of the Up Ramp
- **Up Entries:** Number of entries for the Up Ramp
- **Dwn End:** Final value of the Down Ramp
- **Dwn Entries:** Number of Entries for the Down Ramp
- **Hold Value:** Value of the Hold part
- **Hold Entries:** Number of entries for the Hold part

Figure 10.41 Shape calculator with the display on (Up/Hold/Down example)

Sinc.
- **Num. Entries:** The number of entries in the shape table
- **Num. Lobes:** The number of lobes of the sinc function
- **Shift (Deg):** Shifts the phase of the Sinc function by a value specified in degrees
- **Scale Factor:** Scales the Sinc function (the value specified in this field sets the maximum value of the central lobe)
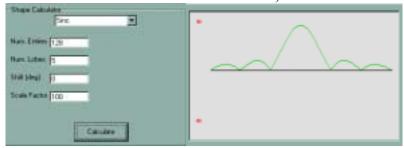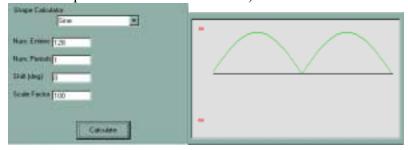
Figure 10.42 Shape calculator with the display on (Sinc function example)

Sine.
- **Num. Entries:** The number of entries in the shape table
- **Num. Periods:** The number of periods of Sine function
- **Shift (Deg):** Shifts the phase of the Sine function by a value specified in degrees
- **Scale Factor:** Scales the Sine function (the value specified in this field sets the amplitude of the Sine function)

Figure 10.43 Shape calculator with the display on (Sine function example)

Gaussian.
- **Num. Entries:** The number of the entries in the shape table
- **Std. Dev.:** Determines the width of the Gaussian shape
- **Scale Factor:** Scales the Gaussian function (the value specified in this field sets the amplitude of the Gaussian function)
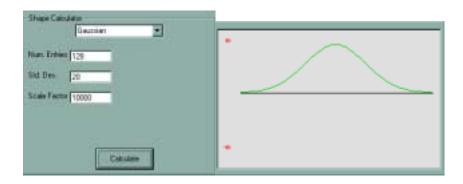
Figure 10.44 Shape calculator with the display on (Gaussian function example)

The **Calculate** button calculates the numerical values of the shape table, minimum, maximum and average values, power ratio with respect to square pulse in % and dB. The **Table contents** window displays the numerical values of the shape table and the **Plot** window displays the calculated and library stored shapes

Table Wizard menu Commands			
File	Exit (Ctrl+Q)		Exits NTNMR table Wizard
Commands	Load table	From Library (Ctrl+L)	Loads table from library
		Calculated (Ctrl+C)	Loads calculated tables
	RF Power Calculation		
	Update Sequence Table List		Updates sequence table list
	Update Library Table List		Updates library table list
	Clear Plot		Clears the Plot Area
Preferences	Allow Plot Overlay		Display multiple shapes in the Plot Area
	Show calculated tables in "Table Contents" box		Toggles on/off display of the numerical values of the table (doesn't work)
	Calculate Sine and Sinc as absolute value		Calculates Sine and Sinc shapes numerical values as positive only

10.7

10.7 Sequence and Table Variables

10.7.1 Sequence Variables

NTNMR allows *unlimited* sequence variables to be defined. With the exception of the Acquisition line, any Event on any line in the sequence can have a variable assigned to it. To create and assign a variable to an Event, double-click

the event (or select Edit… from the Event pop-up) and type "=*variable_name*". The "=" symbol (equal sign) indicates that the text that follows is a variable. The example below shows the steps to create a sequence variable. Sequence variables (displayed in green) appear in the Sequence tab in the Dashbaord. When viewing a sequence, the Dashboard can be opened directly to the Sequence tab by clicking on the **sequence dashboard** button on the Sequence toolbar or by right-clicking on any variable name.

Click on the Event where the variable is to be inserted and select Edit… or double-click on the Event.
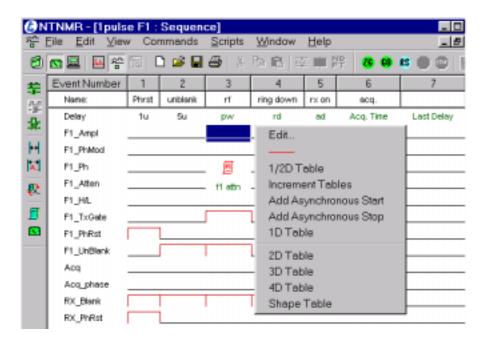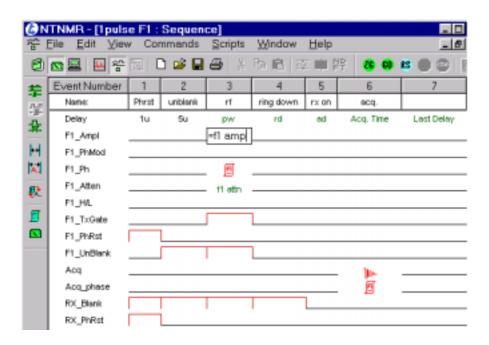
Figure 10.45 Sequence variable drop down menu example
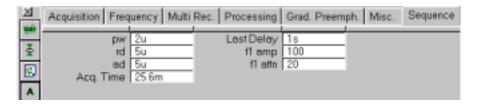
Type the "=" sign followed by the variable name ("f1 amp" in the example).

Figure 10.46 giving a variable name to an event.

Press the [ENTER] key and the variable name will change to green.

The **Sequence** tab in the Dashboard will immediately be updated to include the newly created variable. Clicking on the **Sequence Dashboard** button will open the Dashboard and go directly to the **Sequence** tab as shown below.

Figure 10.47 The Dashboard updated to reflect the addition of the new variable.

Sequence variables in the dashboard can contain mathematical expressions, and can be modified using the variable editor (see chapter 4, Dashboard, for more details).

10.7.2 Table Variables

A variable may also be defined in the same way (using the "*=variablename*" syntax) as the increment for "Auto" tables (see section on tables). This is
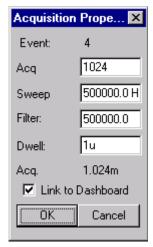
particularly useful when an array of uniformly spaced values is needed, but the spacing depends on some experimental condition (e.g. phase encoding gradient amplitudes, or evolution delays in a 2D experiment). Only the dashboard value needs to be modified to change the experiment. Such variables are called "Table Variables", and are shown in red on the dashboard. For further flexibility, the dashboard field for a table variable may also contain a mathematical expression referring to other variables (see chapter 4, Dashboard, for more details).

10.8 Acquisitions

10.8

The **ACQ** and **ACQ Phase** labels in the timing diagram are always present. This section discusses some important details of setting up the acquisition.

Acquisitions are normally controlled directly from the dashboard. However, an **Acquisition Properties** dialog box (from the popup menu or by right-clicking the icon) is available to set the acquisition parameters from the sequence window:

Figure 10.48 Acquisition properties dialog box.

The Acquisition Properties box allows the user to set the number of points (*all systems)*, sweep width, filter and dwell time (*digital receiver systems only)* for each acquisition event independently. By default, all acquisition events are assigned the number of points, sweep width, dwell and filter settings from the dashboard. Using the Acquisition Properties window allows these parameters to be set on a individual basis. There is also a "Link to Dashboard" check box the can be used to indicate which acquisition events should be connected to the dashboard.

Some systems with digital receivers can also specify a sweep width table for any dimension through the acquisition properties 'Table' button. See below for details.

10.8.1 Acquisition Points

The **number of complex data points** collected during each acquisition event is set by the Dashboard parameter "Acq Points". Multiple acquisition events are allowed in a sequence. The number of points acquired in *all* Acquisition events for one scan is equal to "Points 1D" displayed in the Dashboard. For special acquisition options see the Acquisition Properties section above.

10.8.2 Acquisition Time

The **Dwell Time** is set in the dashboard. Together with the **Acq. Points**, this determines the acquisition time (**Acq. Time**). By default, the duration of the event containing an acquisition is set automatically to **Acq. Time**. This coupling can be disabled in the preferences dialog (see chapter 5).

10.8.3 Acquisition Phase

The acquisition phase for each acquisition is selected from the ACQ Phase line. The acquisition phase table format is exactly like other phase tables. Each acquisition can have a unique phase table, but only phase values of 0, 1, 2, or 3 (x, y, -x, -y). If no ACQ Phase is selected, the default phase is set to 0 (+x).

For a 'CYCLOPS' phase cycle, the table might contain four entries: 0, 1, 2, 3 (or 0, 2, 1, 3 if it is sometimes desired to do only 2 averages). The phase of the transmitter would have a corresponding phase table with four entries to associate transmit pulse phase with acquisition phase. This phase cycle is complete when the scan count ends in a multiple of four. **Use "Soft" Stop to stop a pulse sequence after the next complete phase cycle.**

10.8.4 Acquisition Status

As discussed in the Chapter on the Main NTNMR Window, the acquisition status during an experiment is reported in the **Status Bar** at the bottom of the Main Window (*not* the Data window). The number of scans completed for each dimension and a progress bar report the experiment status. The title bar of the Data window also indicates that an experiment is active and the time the experiment was started.

10.8.5 S.A. Dimension & S.A. Mode Parameters

The S.A. Dimension and S.A. Mode parameters are special dashboard parameters that control the order in which data is stored in the signal averager. Valid entries for S.A. Dimension are 0,1, 2, 3 or 4, while valid entries for S.A. mode are 0 or 1. Note that S.A. Mode is only active for S.A. Dimension = 0.

The S.A. Dimension parameter indicates in which order the data is acquired and stored. The following examples illustrate these parameters:

Example 1: Summation of individual acquisition events inside of a loop

S.A. Dimension = 0, S.A. Mode = 0

This is the default mode for S.A. Dimension = 1. In this mode, each acquisition within a loop will be added to itself each time through the loop. A sequence that contains two acquisition events inside of a loop, each with 512 points, will result in a final data set of 1024 points. The first 512 points will result from the summed data of the first acquisition and the second 512 points will result from the summed data of the second acquisition.

Example 2: Summation of all acquisition events inside of a loop

S.A. Dimension = 0, S.A. Mode = 1

In this mode, all acquisitions inside of a loop will be added. A sequence that contains two acquisitions inside of a loop, each with 512 points, will result in a final data set of 512 points.

Example 3: Block averaging of T1 relaxation data

S.A. Dimension = 2

When acquiring T1 relaxation data is often desirable to acquiring 1 (or several) scans per T1 time point instead of acquiring all averages for a single time point and then continuing on to the next time point. This approach evenly distributes instrumental artifacts over the whole data set.

The classical sequence (inversion recovery) for measuring T1 relaxation is the following pseudo-2D experiment:

    180 - tau - 90 -acq

Where 'tau' is a suitable 2D table of time values. Running this experiment in the non-block averaging mode would involve phase cycling the 180 and 90-degree pulses as well as the acquisition in the "1st" dimension, and writing the 'tau' list as a 2D table. When the experiment is run, one 2D record would be created for each 'tau' value where each record would be the result of "Scans 1D" averages. The 2D record for the first 'tau' would be completed, then the second and so on.

To acquire this data in block average mode, simply write the phase tables for the 180 and 90-degree pulses. For the acquisition as 2D tables, write the 'tau' table as a 1D delay table, and set S.A. dimension to 2. In this case, when the experiment is run, one scan will be acquired for each 'tau' point in turn. Then NTNMR will loop back to the first 'tau' point and acquire a second average for each 'tau' point, etc.

Example 4: Multi-slice imaging

S.A. Dimension = 2

A common use for S.A. Dimension = 2 is a multi-slice imaging experiment. In this case, one average is acquired for each frequency slice indicated in the experiment. After 1 average of all frequencies has been acquired, a second average is then acquired at each frequency and so on.

The experiment then is written so that the frequency selection table is a 1D table and the phase tables for signal averaging are 2D tables. Then when S.A. Dimension is set to 2, NTNMR will acquire 1 scan for each frequency in the 1D table, storing the data in a serial fashion in the signal averager. (Note that if the signal averager was set to 1, the default mode, each of the frequency selected FIDs would be summed.)   Once all entries of all 1D tables have been executed, NTNMR will increment the 2D tables and reset the signal averager memory pointer to the beginning, thus achieving signal averaging.

10.8.5 Multiple Spectral Widths

In a system equipped with a Digital Receiver, it is possible to have multiple acquisitions with independent spectral widths. To accomplish this, the "Link to Dashboard" option is available in the Acquisition Properties dialog. If the check mark is removed, the properties for the selected acquisition are taken from the dialog box instead of the dashboard.

10.8.6 Multiple Receivers

In a system equipped with a multiple digital receivers, it is possible to acquire data in several different modes. In the following discussion, refer to the associated excerpts from NTNMR pulse sequences showing the setup of receivers in a 2-receiver system. The sequence lines have been rearranged to highlight the receiver information.

Individual receiver, normal acquisition.

See Figures 8.49, 8.50. This is the normal single-receiver mode, augmented to allow any one of the installed receivers to acquire (not just the first one). In this mode, all normal features are available: any number of samples up to signal averager memory limit (for dwell times greater than or equal to 1 usec); fast (buffered) mode with up to 4k samples with dwell times below 1 usec; multiple acquisitions on the same receiver; and variable spectral widths (including tables). (Variable SW and fast mode are mutually exclusive). Recycle delays and other limitations are standard.

Data appears in the normal fashion in the signal averager. The dashboard parameter *Points 1D* is the same as *Acq. Points*, (unless there are multiple acquisition events).

Older systems (see above) can only use the first receiver in this mode (Figure 10.49, but not Figure 10.50).
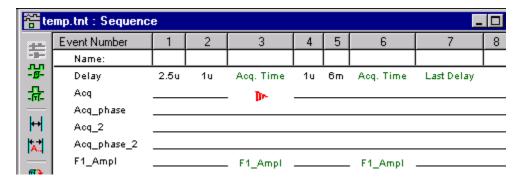
Figure 10.49 Pulse sequence using the first digital receiver to acquire
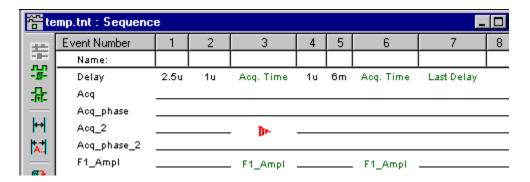
Figure 10.50 Pulse sequence using the second digital receiver to acquire

Multi-receiver, independent acquisition.

See Figures 10.51 and 10.52. In this mode, all normal features are available: any number of samples up to signal averager memory limit (for dwell times greater than or equal to 1 usec); fast (buffered) mode with up to 4k samples with dwell times below 1 usec; multiple acquisitions on the same receiver; and variable spectral widths (including tables). (Variable SW and fast mode are mutually exclusive). The spectral width (*SW +/-*), number of points (*Acq. Points*), etc. can be linked, or independently specified for each receiver. Recycle delays and other limitations must be observed for each receiver, but can in some cases be shorter than for multiple acquisitions on the same channel.

Data blocks appear in the signal averager in the order acquired. E.g. in Figure 10.51, the order is channel 1, channel 2, while in Figure 10.52, it is channel 2, channel 1. The dashboard parameter *Points 1D* is the <u>sum</u> of the *Acq. Points* for all receivers and acquisitions.

This mode is not available with older systems.
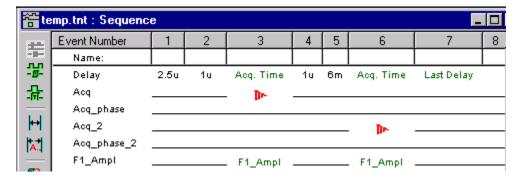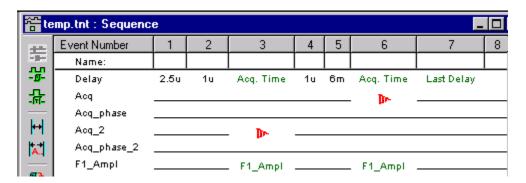
Figure 10.51 Pulse sequence using acquisition on the first digital receiver first
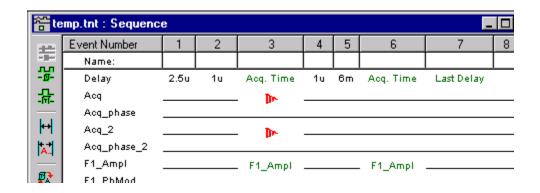
Figure 10.52 Pulse sequence using acquisition on the second digital receiver first

Multi-receiver, simultaneous interleaved acquisition.

See Figure 10.53. This is the original multi-receiver mode. In this mode, only normal acquisitions can be done, and SW tables are not available. The dwell time must be > 1.2 usec * number of active receivers. The bandwidth, number of points, etc. must be identical for each receiver. Recycle delays and other limitations must also be observed for each receiver. Any receiver(s) can be active during the acquisition interval, depending on the presence or absence of an icon on its line. Inactive receivers will be skipped (no data is acquired).

Data appears in the signal averager in interleaved format (Ch1 real, Ch1 imag, Ch2 real, Ch2 imag, etc.). The SortDataPoints command is available to de-interleave the samples. The total number of points (***Points 1D***) is number of active receivers * ***Acq. Points***.

This mode is available on older systems, however, in this case no receivers can be skipped. (For example on a 4-receiver system, acquisitions can occur on 1, 1-2, 1-3, or 1-4, but not 1-2 & 4, or 2-4).

Figure 10.53 Pulse sequence using simultaneous acquisition on both receiver channel.

Multi-receiver, simultaneous non-interleaved acquisition.

 This is an "unofficial" mode, and may change in a future implementation. However, it may be useful in some cases, as when the required bandwidth is larger than available in the interleaved mode. This mode uses the "fast mode" buffer, and the on-board FIFO. Multiple spectral widths (in separate acquisition events) and SW tables cannot be used. The maximum data points per channel that can be reliably acquired is 1k (limited by the FIFO, not the DSP memory). The spectral width (SW +/-), number of points (Acq. Points) can be different for each receiver, but the acquisition times (Acq. Time) should be identical. Thus the bandwidths should have simple relationships. For example, channel 1: 1k points at 500nsec/pt; channel 2: 256 points at 2usec/pt.

Data blocks appear in the signal averager in channel-number order. The dashboard parameter *Points 1D* is the sum of the *Acq. Points* for all receivers and acquisitions.

To use this mode, add the parameter "NoInterleave = 1" in Apollo.ini under the various DigitalReceiverG2 objects. The parameter defaults to 0 (interleaved simultaneous acquisitions).

This mode is not available on older systems.

10.8.7 Single-Point Acquisitions

Analog (ADC) Systems

Single-point acquisition is initiated by the following actions:
- Set dashboard 'Acq. Points' = 1
- Place an acquisition event at the point where the samples are to be acquired (no special icon). Acquisition phase can also be set in this event.
- Place 'loop start'/'loop end' events in the sequence bracketing the acquisition. The loop start can be the same event as the acquisition. Loop end should not be on the acquisition event - the compiler generates the wrong total number of points (only 1 !) for the receiver in this case.
- Set the loop counter as required for the total number of points. The loop counter can be a variable "=Points 1D" for convenience in using the standard dashboard field.

- Set the dashboard 'Filter' to a valid bandwidth about 25% (or more) larger than the actual dwell (see 'restrictions' for more details). Filter must be set after SW, since the dashboard always resets the filter to match the SW when the latter is changed.

Digital Receiver Systems

In this discussion, the term 'samples' to refers to actual ADC samples. 'Points' refers to the output values going to the signal averager and displayed by NTNMR. The current implementation switches the ADC clock on and off in bursts. Each acquisition trigger generates a burst of samples appropriate for the selected filter bandwidth. After down conversion and decimation, the burst is reduced to a single complex point, which is sent to the signal averager. At the beginning of the scan, the normal filter group delay occurs - that is the first few points are nearly zero. At the end of the scan, additional ADC samples are forced to flush out the pipeline and get the last points. All of the samples go through the same digital filter. Thus, there is a 'memory' effect from one point to the next (just as in a normal acquisition with the same bandwidth). However, anything that happens in between is not sampled and hence not 'remembered'. This should work fine for situations such as CRAMPS, MLEV, etc. It is more problematic for a situation like SPRITE where the points could be completely independent in phase as well as amplitude.

Single-point acquisition is initiated by the following actions:
- Set dashboard 'Acq. Points' = 1
- Place an acquisition event at the point where the samples are to be acquired (no special icon). Acquisition phase can also be set in this event.
- Place 'loop start'/'loop end' events in the sequence bracketing the acquisition. The loop start can be the same event as the acquisition. Loop end should not be on the acquisition event - the compiler generates the wrong total number of points (only 1 !) for the receiver in this case.
- Set the loop counter as required for the total number of points. The loop counter can be a variable "=Points 1D" for convenience in using the standard dashboard field.
- Set the dashboard 'SW+/-' to correspond to the actual dwell time (for processing purposes). For convenience, math expressions can be used to construct the correct sequence timing from the dashboard-specified 'SW' and any other quantities (e.g. rf pulse duration).
- Set the dashboard 'Filter' to a valid bandwidth about 25% (or more) larger than the actual dwell (see 'restrictions' for more details). Filter must be set after SW, since the dashboard always resets the filter to match the SW when the latter is changed.

- Turn off the preference for 'Automatically set delays for Acq.', so that the delay for the acquisition event will not be overridden by the dashboard dwell time.

There are some restrictions on single-point acquisitions for the digital receiver:
- Total number of points per 1D scan must be at least as many as for a normal scan with the same filter bandwidth. Otherwise, no interesting data will come out of the filter.
- Acquisition phase cannot be a 1/2D table (incremented in loop). This is partly because the 1/2D mechanism was never implemented for G1-based event modules, and partly because the results would be garbage anyway.
- Actual dwell time between points must be a multiple of 400ns. Otherwise, the phase of successive samples does not agree, and they tend to cancel each other out.
- Absolute minimum dwell time (refer to hardware reference). Minimum dwell also depends on the filter bandwidth since a decimating filter is used, and is approximately 25% longer than the dwell corresponding to the filter bandwidth. There is no upper limit to the dwell time.
- Filter bandwidth selections are limited (refer to hardware documentation for allowable bandwidth selections).

10.9

10.9 Real Time Adjustment Mode

The Real Time Adjustment mode allows all user defined sequence parameters and virtually all Dashboard parameters to be interactively adjusted while acquiring data. Clicking on the ⬚ button in the Dashboard activates the real time adjustment mode. The Adjustment dialog is added to the Dashboard on the right hand side of the window as shown below.
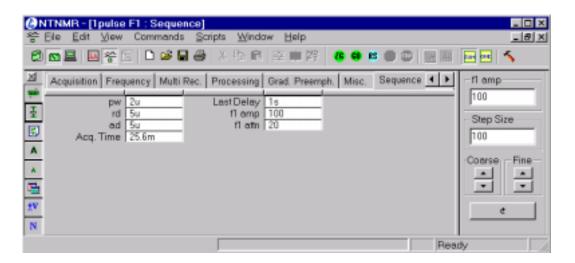
Figure 10.54 Real Time adjustment mode window

To activate a parameter for adjustment, left-click in the desired parameter's text edit field. The adjustment dialog will be updated, inserting the new parameter name, it's current value and a default **Step Size** that is 10% of the current value. The **Step Size** can be manually edited as in the example above where it has been set to 100.

When a repetitive scan (**RS**) is activated, the Real Time Adjustment Mode is active. Clicking on the up and down arrows in the **Coarse** field will adjust the parameter by the amount indicated in the **Step Size** field. Clicking on the up and down arrows in the **Fine** field will adjust the parameter by (0.1 * **Step Size**). Once an adjustment has been made the Toggle button at the bottom of the window is updated to display the starting value. Clicking on the Toggle button will re-set the parameter to the starting value and update the Toggle button to display the last value used. Each time the Toggle button is clicked the *last value set* will be recalled. This allows the user to toggle between values with a single click.

Real Time Adjustment mode is only available when repetitive scan mode is active. Some Dashboard parameters are not available for Real Time Adjustment, such as text and information only fields.

10.10 Multi-Dimensional Sequences

10.10

10.10.1 Overview

Multi-dimensional acquisitions of up to four dimensions are easily facilitated with NTNMR.

NTNMR employs an implicit loop structure to implement multi-dimensional experiments. A simple pulse-and-acquire experiment is a "one-dimensional" (1D) experiment. When signal averaging is employed, the sequence repeats for each average (or scan or repetition) (up to the dashboard limit "Scans 1D"). This is automatic – no loop event is required – and the implicit loop goes from the last event to the first event.

In multi-dimensional experiments, the same mechanism is used – a dashboard value ("Points 2D", "Points 3D", etc.) specifies the size of the data set in the 2nd and higher dimensions. At the completion of a 1D portion of the experiment, the data is stored (a "1D Record") and the 2D counter ("Actual Points 2D") is incremented. The experiment continues, with the data for the next 1D record being stored separately. Again, the loop is automatic, from the last event to the first event.

In order to modulate the signal in the second dimension, some variable in the sequence must change from one 1D record to the next. (In 2D spectroscopy experiments, this is typically a delay, while in imaging, it is typically a gradient amplitude). The mechanism provided for this is a "2D table" (more generally, an "nD table"). The table index is automatically incremented along with the 2D counter.

Tables can be specified from 1 to 4 dimensions and each dimension in the sequence is displayed in a new section of the pulse sequence editor. The example below shows the addition of a 2D-Delay table to a sequence.

10.10.2 A Simple Example

The following starting point for a COSY sequence was created using the general techniques discussed above. It is a 1D sequence with two RF pulses. The delay between the pulses, event number 4 (named "tau") has been assigned a value of 1u.
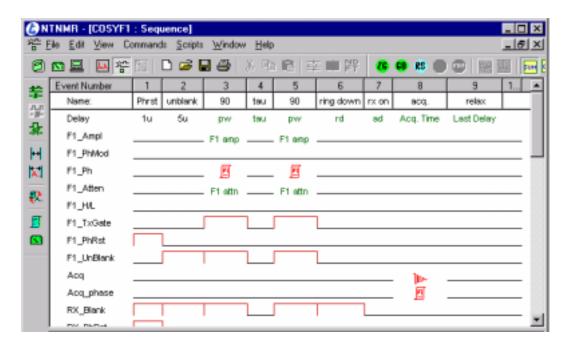
Figure 10.55 COSY sequence

A 2D table was added for the Delay line in Event 4. When a multi-dimensional table is added, a new section is created for that dimension at the bottom of the sequence. Lines are added to the new multi-dimensional section of the sequence only as needed and they are named in the format *linename_dimension*. The example below shows the line "Delay_2D", indicating that this new line is associated with the "Delay" line in the main sequence window (i.e., the first dimension). All 2D tables that are created appear in the 2D section of the sequence. New lines are added as needed.
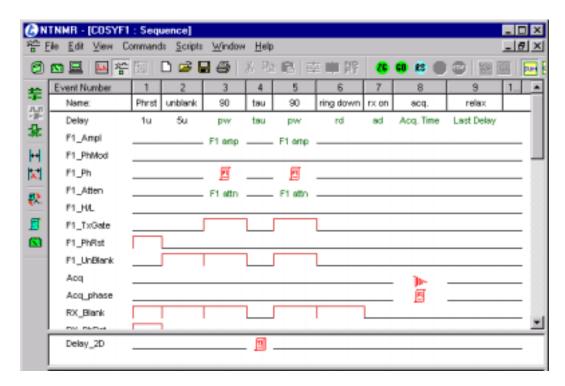
Figure 10.56 COSY sequence with 2D table

Multi-dimensional tables are edited in the same manner as their 1D counterparts; right-click on a table or click the Edit Tables button to activate the Edit Tables window. In this case, we would use the "Auto" feature, and select the "use Dwell Time" check box to link the delay to the dashboard.

Multi-Dimensional Table Concepts

Multi-dimensional tables in NTNMR function in following way. The highest dimension table modifies the next lowest dimension table. This means that a 2D table *will* modify the value set in the corresponding 1D event, whether it is a table or a single value. Similarly, a 3D table will modify a 2D Event and a 4D table will modify a 3D Event. Take for an example the sequence above. Event 4 of this Cosy sequence has a 1D "tau" value and a 2D delay table. The 2D delay table that appears in the "Delay_2D" line will modify the 1D "tau" value in Event 4. Every time the 2D counter is incremented (i.e. for each 2D point) the next value in 2D table will be added to the 1D "tau" value. In most cases the higher dimensional table value will add to the value in the next lowest dimensional table.

There are a couple exceptions to this concept. Multi-dimensional frequency tables associated with "F1_Freq, F1_Freq_I, F1_Freq_Q…." do not modify the next lowest dimension table. The higher dimensional table values are used and the lower dimensional values are ignored. This will probably change in later

version of NTNMR   Multi-dimensional tables involving gradient amplitudes modify differently then the others. The gradient amplitude table values will multiply as a percentage. Take for example a 1D gradient amplitude value of 10, and a gradient amplitude 2D table of –100, -50, 50, 100. These values are considered as percentages. The first record would have gradient amplitude value of –10, the next record would be –5, the third record would be 5 and the last record would be 10.

 All tables that are set to the **Auto** mode, such as Delay and Loop tables, will modify the corresponding 1D Event. Consider the basic COSY example shown above. If the Delay_2D table in Event 4 was set to be auto-incremented (**Auto** mode) based on the Dashboard Dwell_2D parameter, the time for Event 4 during acquisition of the first 2D point would be 1u (the 1D Event time). Each 2D point after that will be the accumulation of the Event time for the previous 2D point and the Dashboard parameter Dwell_2D. So, for the second 2D point, the time for Event 4 would be (1u + Dwell_2D). For the third 2D point, the time would be (1u + Dwell_2D + Dwell_2D) and so on.   The same logic is applied to Loop tables that are auto-incremented.

Phase tables *always* modify their corresponding (dimension - 1) Events. The extension of the basic COSY example above to a phase sensitive COSY will illustrate this point. First consider the TPPI case.
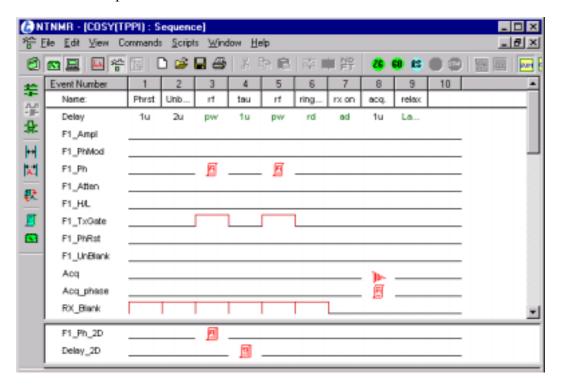
Figure 10.57 COSY(TPPI)

In the sequence shown above, the basic COSY experiment has been modified to add a 2D-phase table. This table, which appears in the OBS_PH_2D line, modifies the 1D phase table that is in the OBS_PH line of the sequence. The tables have the following entries:

- 1D phase table: 0 2 2 0 1 3 3 1
- 2D phase table: 0 1 2 3

The 2D table is incremented with respect to the 2D scan counter (i.e. with each 2D point) and the entry that is valid for the current 2D point is added to *all entries* in the 1D table. Always keep in mind that 2D tables (also 3D and 4D) are cycled in exactly the same way as 1D tables, except that they are cycled with the second dimension counter (third or fourth for 3D and 4D tables), instead of the first. The following table illustrates what the phase cycling would be for the above example for the first 5 2D points.

2D Point	phase cycle
1	0 2 2 0 1 3 3 1
2	1 3 3 1 2 0 0 2
3	2 0 0 2 3 1 1 3
4	3 1 1 3 0 2 2 0
5	0 2 2 0 1 3 3 1

This achieves the required 90 incrementation on the "preparation" pulse with respect to the 2D point being acquired; that is required for TPPI.

The following, slightly more complicated example, illustrates the implementation of a phase sensitive COSY experiment via the mixed States-TPPI method.
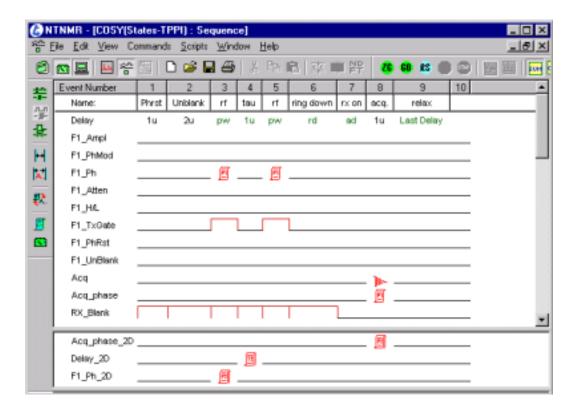
Figure 10.58 COSY(states-TPPI)

This sequence adds another 2D-phase table which appears in the "Acq_phase_2D" line. This table modifies the "Acq.phase" table. The following phase tables are used in the example above:

Transmitter Phase Tables
- OBS_PH "preparation" pulse (Event 3): 0 2 2 0 1 3 3 1
- OBS_PH_2D (2D Event 3): 0 1 2 3

Acquisition Phase Tables
- Acq_phase (Event 8): 0 2 2 0 1 3 3 1
- Acq_phase_2D (2D Event 8): 0 0 2 2

Additionally, the Delay_2D table (2D, Event 4) is set to **auto-increment** using the Dashboard parameter **Dwell_2D** with the **Increment Scheme** set to "Every 2 Passes". (See the section on tables for details on these table settings)

The following tables show the Phase Cycling and 2D delay with respect to the 2D point number:

2D Point	2D Delay (Event 4)	Transmitter Phase Cycle	Acq. Phase Cycle
1	1u	0 2 2 0 1 3 3 1	0 2 2 0 1 3 3 1
2	1u	1 3 3 1 2 0 0 2	0 2 2 0 1 3 3 1
3	1u + Dwell_2D	2 0 0 2 3 1 1 3	2 0 0 2 3 1 1 3
4	1u + Dwell_2D	3 1 1 3 0 2 2 0	2 0 0 2 3 1 1 3
5	1u + (Dwell_2D * 2)	0 2 2 0 1 3 3 1	0 2 2 0 1 3 3 1
6	1u + (Dwell_2D * 2)	1 3 3 1 2 0 0 2	0 2 2 0 1 3 3 1
7	1u + (Dwell_2D * 3)	2 0 0 2 3 1 1 3	2 0 0 2 3 1 1 3
8	1u + (Dwell_2D * 3)	3 1 1 3 0 2 2 0	2 0 0 2 3 1 1 3

10.11

10.11 Pulse Sequence Configuration File

This section is provided for the advanced user and is not necessary for understanding pulse sequences in NTNMR. You may also wish to consult the hardware manual for more information.

10.11.1 Overview

The ***config.con*** file is a text file used by NTNMR to format the Pulse Sequence Window. This file must be located in the 'Config' folder in the NTNMR folder. This file contains the interface information necessary to assign output lines from the Tecmag pulse programmer for spectrometer control functions. In general, **this file should not be edited by the user**. However, there may be cases, such as the construction of a home built spectrometer based on a Tecmag Pulse Programmer, where it may be necessary to edit the ***config.con*** file.

10.11.2 File description

The following is an example if a ***config.con*** file.

```
;file name = config.con
;
;    Definition of LP lines for Apollo HF system (2 channel + gradients)
;    Consistent with G2 and multi-receiver systems
;
;     -- File Format --
;
; <1> = Name
; <2> = Module Prefix
; <3> = Start Address
; <4> = Number of Bits
; <5> = Minimum delay
; <6> = Maximum Delay
; <7> = delay Unit
; <8> = Icon type
; <9> = Visible
; <10> = Private Data
; <11> = Group Number
;
; Any text after ";" on each line is ignored.
; Fields are delimited by either spaces or tabs.
;
; <1>          <2> <3><4><5><6><7> <8><9><10><11>
F1_Ampl       EA1 11 12  1  0  0   R2  1  0    6  ; F1 amplitude modulation
F1_PhMod      EP1 11 12  1  0  0   P2  1  0    1  ; F1 phase patterns
F1_Ph         BP2 05  2  1  0  0   PH  1  0    0  ; F1 phase cycle (90/180)
F1_Atten      EA1 22  7  1  0  0   A2  1  5    6  ; F1 atten - note private data!
F1_H/L        EA1 23  1  1  0  0   TX  1  0    6  ; F1 user attenuator
F1_Freq           DD1 23 24  0  0  0   O1  1  0    0  ; F1 frequency control
F1_TxGate     EA1 31  1  0  0  0   TX  1  0    6  ; F1 transmitter gate
F1_PhRst      BP2 00  1  0  0  0   TX  1  0    0  ; F1 phase reset
F1_UnBlank    BP2 13  1  0  0  0   TX  1  0    0  ; F1 amplifier unblanking


Acq           AQ1 23 24  0  0  0   ACQ 1  0    0  ; Acquistion ch 1
;Acq_2        AQ2 23 24  0  0  0   ACQ 1  0    0  ; Acquistion ch 2
Acq_phase     AQ1  3  2  0  0  0   PH  1  0    0  ; Acquisition phase ch 1
;Acq_phase_2  AQ2  3  2  0  0  0   PH  1  0    0  ; Acquisition phase ch 2


RX_Blank      BP1 01  1  1  0  0   TX  1  0    0  ; RX blank ch 1
;RX_Blank_2   BP2 14  1  1  0  0   TX  1  0    0  ; RX blank ch 2
RX_Freq       DD0 23 24  0  0  0   O1  1  0    0  ; Receiver LO 1 frequency control
RX_PhRst      BP1 00  1  0  0  0   TX  1  0    0  ; Receiver LO 1 phase reset


Gr_Shape      EG1 31 16  0  0  0   SH  1  0    0  ; Read gradient shape
Gr_Amp        EG1 15 16  0  0  0   G2  1  0    0  ; Read gradient amplitude
Gp_Shape      EG2 31 16  0  0  0   SH  1  0    0  ; Phase gradient shape
Gp_Amp        EG2 15 16  0  0  0   G2  1  0    0  ; Phase gradient amplitude
Gs_Shape      EG3 31 16  0  0  0   SH  1  0    0  ; Slice gradient shape
Gs_Amp        EG3 15 16  0  0  0   G2  1  0    0  ; Slice gradient amplitude
Grad_Sync     EM1 00 01  0  0  0   TX  1  0    0  ; Gradient reset



; External trigger is on EM1 for gradient system, BP1 otherwise
;Ext_Trig     EM1 31  1  0  0  0   ET  1  0    0  ; External trigger (dummy LP line)
;Ext_Trig     BP1 31  1  0  0  0   ET  1  0    0  ; External trigger (dummy LP line)
```

```
 Ext_Trig      EM1 31  1  0  0  0   TX  1  0   0  ; External trigger (dummy LP line)
 ;Scope_Trig   BP1 06  1  0  0  0   TX  1  0   0  ; Scope trigger for diagnostics
 Scope_Trig    BP1 19  1  0  0  0   TX  1  0   0  ; Scope trigger for diagnostics
 Lock_Blank    BP1 20  1  0  0  0   TX  1  0   0  ; Blank lock during grad pulse
 RSM_Store     EM1 12  1  0  0  0   TX  1  0   0  ; RSM - set rotor phase
 RSM_Read      EM1 13  1  0  0  0   TX  1  0   0  ; RSM - sync with rotor phase
 RSM_Mode      EM1 14  1  0  0  0   TX  1  0   0  ; RSM mode vs. Ext trig mode

 ;XVT_Tx_Blank BP2 02  1  0  0  0   TX  1  0   0  ; Transverter LO blanking

 ;XVT_Rx_Blank BP2 03  1  0  0  0   TX  1  0   0  ; Transverter LO blanking
```

The semi-colon character is used to comment on a line. Everything after the ";" on any line is ignored. Individual lines can be commented out. (As in the "ATN2" and "SPOIL" lines in the example above)

NTNMR creates the sequence window with the lines in the order that they appear in the 'config.con' file. This is only the default order and lines can be repositioned in the sequence window.

The fields in the file are labeled as <1>, <2>, … <8> for informational purposes only. The fields are described below in reference to the labels in the above example.

<1>   Line Name. This is the exact text that will appear in the pulse sequence window when NTNMR is run.

<2>   Module Prefix. This tells the software what type of module this line in the pulse sequence is controlling. There are several module types: EM (*event module*), ES (*event module slave*), GM (*gradient module*), etc. Each module has a number assigned to it, EM1, IO3, GM4, etc. Consult the hardware documentation for more information about the various module types and their function.

<3>   Start Address. This is an integer number that defines the starting address for the pulse programmer lines to use for the current line in the pulse sequence.

<4> Number of Bits. This is an integer number that defines the number of pulse programmer lines to use starting from the Start Address. For example, in the 'config.con' example above, the line Gx_Shape with a starting address of "31" and a number of bits value of 16. This means that there are 16 bits assigned to the control of the GM1 module. The 16 bits are 31, 30, 29, …16.

<5>, <6>, <7>   These fields are currently unused.

<8> Icon Library. This field contains a two-letter code that assigns the type of icon to display in the NTNMR graphical sequence display. The various icon type and their function are discussed in the next section.

<9> Visible. This flag determines whether or not the defined line will be visible in the graphical pulse sequence editor in NTNMR. 1 = visible, 0 = invisible.

<10> Private Data. This field is used for various line-specific data assignments. For example, this field may be used to indicate the resolution of the transmitter phase shifter, the resolution of the transmitter attenuator, etc.

<11> Group Number. It is designated for asynchronous grouping. The lines specified with the same group number will be executed asynchronously once they are activated in the pulse sequence.
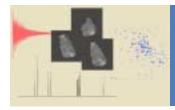
In the Pulse Sequence Window, the **Delay**, **ACQ**, **ACQ Phase** and **Loop** row labels are added to the timing diagram automatically by the software. They are not part of the 'config.con' file.

10.11.3 Icon Libraries

Each line used in config.con has an 'icon library' or line type associated with it. The type specifies what menu options will appear in the popup menu, and what icons will represent the selections in the graphical window. For specific examples of the icons that appear in the pulse sequence window, see the section above entitled "Sequence Elements." The table below is a partial list of icon types defined in NTNMR.

Library	Icons	Information
TX	Gate on, Gate off	Used for controlling rf transmitter channels, attenuators and external devices.
PH	Fixed phases (+x, +y, -x, -y) and 1 - 4D tables	Used for controlling rf phase shifts and for signal averager phase cycling.
ACQ	Acquisition	Indicates the start of data acquisition by the signal averager.
SH	Shape	Allows the creation of waveform (i.e. shape) tables for gradients or for RF pulse shaping.
GR	Gradient	Gradient amplitude control.
PS	Shape	Phase modulation.
RM	Shape	Transmitter amplitude modulation.
AT	Shape	Transmitter attenuator.
ET	Gate on, Gate off	Activate the external trigger.

LD	Shape, gate	Used to assign a group of lines to be used for clocking out a table of values one value at a time.
R2	Shape	Transmitter amplitude modulation with asynchronous and ½ D table options.
P2	Shape	Phase modulation with asynchronous and ½ D table options.
A2	Shape	Transmitter attenuator with asynchronous and ½ D table options.

11 – Tools & Accessories

This chapter describes six software tools. The first four, Sample Interface control, RF Control, Tecmag Lock and Shims are meant to be used with tecmag's optional accessories. The last two, NMR data Manipulation and NMRWizard can be used with all Tecmag products.

11.1 Sample Interface Control

11.1.1 Introduction

The Sample Interface Control Program is used to control the sample temperature and spin rate for systems equipped with a VT Control Unit and/or Air Control Unit. To start the program, go to the Windows Start Menu, Tecmag program group, or choose **SampleInterface** in the NTNMR Script menu. The associated preferences file needed for this program is: *SampleInterface.ini*.

11.1.2 Menu Bar

Figure 11.1 Sample Interface Control menu bar

The Menu Bar includes a **Commands** menu, **Preferences** menu and **Probes** menu in addition to the standard **File** and **Help** menus.

11.1.3 File Menu

Figure 11.2 Sample interface **File** menu

The **File** Menu has only one entry.

Exit	Exit the Sample Interface program. The log file (if any) will close, and preferences will be saved. Independent air controllers and VT controllers continue to operate. Closed loop control *with the host PC as controller* is disabled.

11.1.4 Commands Menu

Figure 11.3 Sample interface **Commands** menu

The **Commands** menu allows the user to perform some operations immediately.

Read **All**	Forces an immediate read of spin rate and/or VT temperature when Continuous mode is disabled. Ctrl+A performs the same operation
Reset Hardware	Reset any hardware devices. Useful if power has been removed from the controller(s).

11.1.5 Preferences Menu

Figure 11.4 Sample interface **Preferences** menu

The **Preferences** menu is used to configure the operation of the program. The menu items are 'toggles': A check mark beside a menu option indicates that the feature is enabled.

Continuous Update	Allows the program to automatically and periodically update spin and temperature readings.

Stay on <u>T</u>op	Prevents the Sample Interface windows from being hidden by other programs.
Spin Enabled	Enables operation of a spin controller. Turn this off when not in use to improve response of program.
VT Enabled	Enables operation of a variable temperature (VT) controller. Turn this off when not in use to improve response of program.
Log Enabled	Enables logging of data. Entries are made in the specified file (default ntnmr\vtlog.log) consisting of <time stamp, temperature (deg C), spin rate (Hz)>.
Show Diagnostic Window	Displays the diagnostic window, which shows messages from the controllers and permits setting up the closed loop (PID) parameters.
Strip Chart Display	Turns on the strip chart display of temperature and spin rate.

11.1.6 Probes Menu
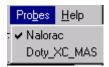
Figure 11.5 Sample interface **Probe** menu

The **Probes** menu allows the user to indicate the probe in use. A check mark indicates the currently selected probe. The entries in this menu are system-specific. The probes must have been previously defined in the preferences file (SampleInterface.ini), with information concerning the VT and air control modes, control loop parameters, etc. Contact Tecmag for details.

11.1.7 Help Menu

Figure 11.6 Sample interface **Help** menu

The **Help** Menu has only one entry.

About	Display version and copyright information for the program.

11.1.8 Status Bar

Figure 11.7 Sample interface Status bar

Located at the bottom of the main window and displays the actual temperature, actual spin rate and actual VT air status (Fault or OK).

11.1.9 VT Control Tab
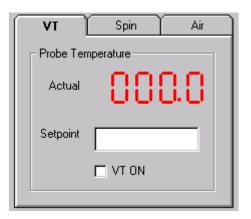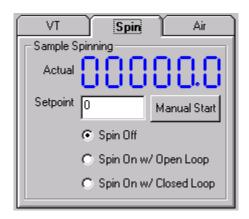
Figure 11.8 VT Control tab

Click on VT tab on main window to activate it. Type in the required set point for VT temperature in °C and hit return. Check the VT On box to turn on the VT heater and also the VT air if it was not already on. Uncheck it to turn off the heater and VT air will stay on unless VT air check box is unchecked in Air Control Tab.

11.1.10 Spin Control Tab
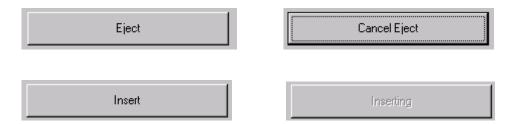
Figure 11.9 Spin Control tab

Click on Spin tab on main window to activate it. Type in the required set point for spin rate in Hz and hit return.

There are three option buttons:

- **Spin Off**: stop spinning sample and spin air will be turned off. The check box of spin air in Air tab will be changed correspondingly.
- **Spin On with Open Loop**: begin spinning sample by using the lookup table values defined in initialization file. Spin air will be turned on simultaneously and the check box of spin air in Air tab will be changed correspondingly.
- **Spin On with Closed Loop**: begin spinning sample by using PID corrector parameters that can be changed in Diagnostic window. Spin air will be turned on simultaneously and the check box of spin air in Air tab will be changed correspondingly.

Figure 11.10 Sample Insertion control buttons

Click on Eject command button to start ejecting the sample. The caption "Eject" will change to "Cancel Eject" during the process of ejecting the sample. Once ejecting is finished the caption will change to "Insert". Click on the

button when the sample is ready to be inserted into probe. The caption "Insert" will change to "Inserting" during the process of inserting sample and the button will be deactivated to avoid interfering.

11.1.11 Air Control Tab
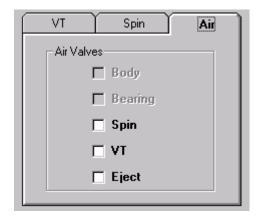
Figure 11.11 Air Control tab

Click on the Air tab in the main window to activate it. Check the spin VT or Eject box to turn on the corresponding valve; uncheck it to turn it off. Some checkboxes may be disabled in the initialization file depending on the hardware in Air control box. Also, turning on the Spin valve will automatically turn on the Bearing valve and vice versa.

11.1.12 Strip Chart Display window

Figure 11.12 VT and Spin Rate strip charts

Activate the window from **Preferences** menu. The red picture box is for VT temperature display and blue is for spin rate display. The current time and date at the last point of recording is shown on both displays.
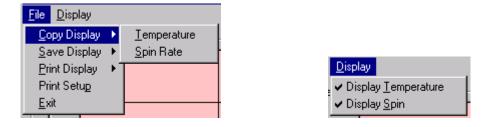
Figure 11.13 VT and Spin Rate strip charts drop down menus

Under **File** menu one of both display picture boxes can be copied to clipboard, saved as a bitmap file or printed after print setup. One or both picture boxes can be displayed by **Display** menu.
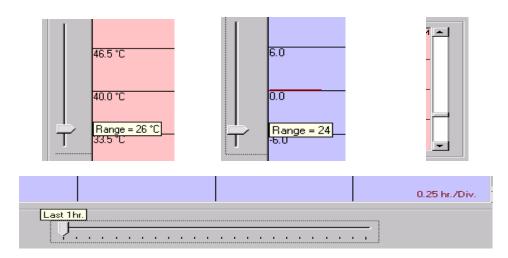
Figure 11.14 VT and Spin Rate strip charts Sliders

There is a vertical Slider bar on the left of each display box to change the display range of the "y" scale and the tool tip will be displayed when the mouse is placed over the slider bar which will indicate current range. There is a vertical Scroll bar on the right of each display box (only one shown here) to change "y" offset. There is one horizontal slider bar for both displays to change display range of x scale and tool tip text indicates the current value. Note that the value per division on "x" scale is changed accordingly.

11.1.13 Diagnostic Window
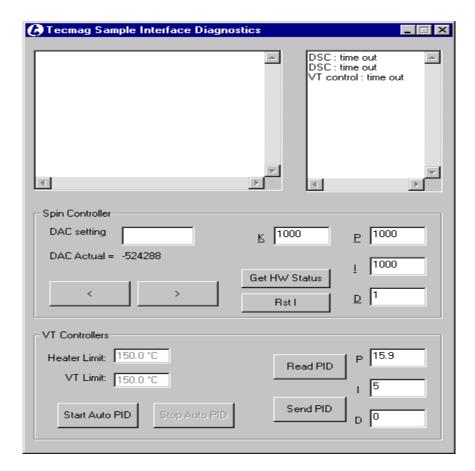
Figure 11.15 Sample interface diagnostic window

Mostly used by Tecmag technical personnel or a system manager to setup the hardware and configure the software accordingly. **User should use caution in making changes within this window.**

11.1.14 The Modification of the Initialization File

Initialization file *SampleInterface.ini* may be modified before starting the program. This should be done with great care and caution, and under the guidance of Tecmag technical personnel.

11.2 RF Control

11.2.1 Introduction

The RF Control Program is used to control the Apollo NMR RF system including the probe interface and UHF/SHF transverter. To start the program, go to Start, Program and Tecmag group, or choose **RFControl** in the NTNMR Script menu. Associated files needed for this program include initialization file (RFControl.ini), customized nucleus list file (*nucleus.txt*) and, optionally, tuning files (*tune.tnt* and/or *sweep.tnt*).

11.2.2 Menu Bar

Figure 11.16 RF control menu bar
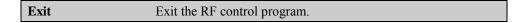
The Menu Bar includes a **Preferences** menu and **Nucleus** menu in addition to the standard **File** and **Help** menus.

11.2.3 File Menu
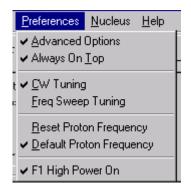
Figure 11.17 RF control **File** menu

The **File** Menu has only one entry.

Exit	Exit the RF control program.

11.2.4 Preferences Menu
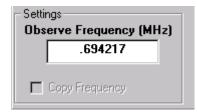
Figure 11.18 RF Control **Preferences** menu and frequency setting display

The **Preferences** menu is used to setup/configure the program.   The menu items are 'toggles': a check mark beside a menu option indicates that the feature is enabled.

Advanced Options	Shows Settings frame, which shows the current observe frequency on NTNMR dashboard and new observe frequency can be typed in here and updated on NTNMR dashboard. Uncheck it to remove the frame from displaying in window.
Always On Top	Keeps the program window always on the top of the other windows.
CW Tuning	Chooses CW Tuning mode. Checking it will uncheck Freq Sweep Tuning.
Freq Sweep Tuning	Chooses Freq. Sweep Tuning mode. Checking it will uncheck CW Tuning.
Reset Proton Frequency	If a customized proton frequency is chosen on the NTNMR dashboard, Reset Proton Frequency has to be checked to allow the RF Control program to be updated accordingly. This is particularly important before you use the Nucleus menu to select different nucleus and the corresponding frequency.
Default Proton Frequency	If the default proton frequency of NTNMR (Preferences in Edit menu) is used, then the Default Proton Frequency is checked.
F1 High Power On	Proton high power amplifier(s) can be turn on or off for channel(s).
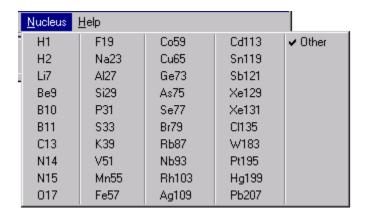
Figure 11.19 RF Control Nucleus menu

The **Nucleus** menu can be customized from the list in *nucleus.txt* file. Checking any one of them will uncheck the others. The NTNMR dashboard will be updated accordingly for nuclear and corresponding frequency. If any others are checked, "undefined" will be shown in Nucleus field of the NTNMR dashboard.

11.2.5 RF Channel Selection Buttons

Figure 11.20 RF Channel selection buttons

The number of available channels is defined in *RFControl.ini* file. Clicking on each channel button will depress the button and release the others except for the tune button (the one with a curve shape).

Probe tuning process:

The tune button is independent from the other channel buttons. Toggle the button to start or stop the tune process. Once the tune button is depressed the following warning message will be displayed to make sure the acquisition of tuning process is only controlled by RF Control program. Particularly, DO NOT stop the acquisition from NTNMR STOP command button but click on the tune button to stop it or close the tuning window to stop it.
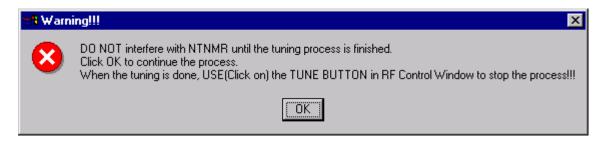
Figure 11.21 Tuning warning message

Click OK to continue and the pre-created tune file will be loaded in NTNMR and start the tune process automatically. According to the tune mode selection made in Preferences menu (CW Tuning or Freq. Sweep Tuning), either *tune.tnt* or *sweep.tnt* will be loaded for tuning process. (Note: depending on hardware the sweep tuning mode may not be always available. And the Freq. Sweep Tune menu item can be disabled by initialization file. Even both tune modes may not be available without probe interface box.) Once the tune process is started the LED meter on the F1 probe interface is filled up. Then connect the appropriate channel of the probe to the BNC connector labeled "Tune" next to the LED meter.

- For **CW Tuning process**:
  The following CW Tuning window will pop up on screen.
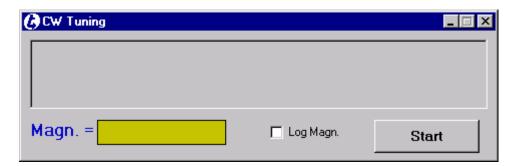
Figure 11.22 CW Tuning window before tuning is started

Click on **Start** button to start the CW tuning process and the caption of the button will then be changed into **Stop**. The tuning will go on until you hit the **Stop** button.
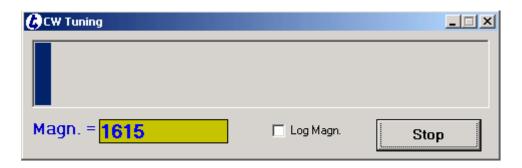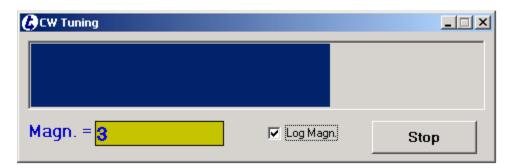
Figure 11.23 CW Tuning window after tuning is started

CW tuning result is displayed as a progress bar based on magnitude value. The magnitude value is shown in a yellow text box. The Log Magn. check box is used to select magnitude scaling mode and can be set at any time.

Figure 11.24 CW Tuning window with tuning in progress and the Log Magn box on.

To tune the probe, simply adjust it to minimize the number of LED's lit on the meter while minimizing the progress bar on screen.

The tuning progress bar will only be displayed during the process. The initial maximum of the progress bar scale is set based on initialization file and can be changed by hitting SHIFT and < and > keys on keyboard; the new maximum will be saved in initialization file and recalled next time starting the program. The Home key on keyboard can be used to restore the original maximum value saved in initialization file. However, the keys are only activated during the tuning process and the form has focus.

- For **Freq. Sweep Tuning process**:
  The following RF Sweep Tuning window will pop up.
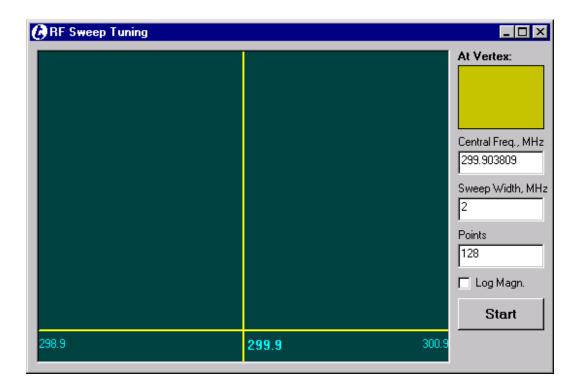
Figure 11.25 RF sweep Tuning window

Click on **Start** button to start the frequency sweep tuning process and the
caption of the button will then be changed into **Stop**. The tuning will go on
until you hit the **Stop** button.
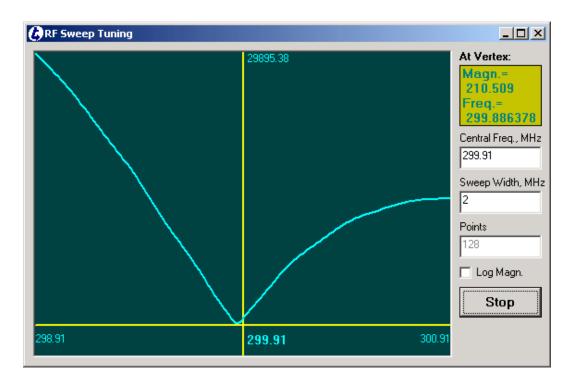
Figure 11.26 RF sweep Tuning window with tuning started

Frequency sweep tuning result is displayed as a curve with magnitude as y-axis and frequency as x-axis. The "origin" point has magnitude of zero and center frequency (observe frequency on the dashboard of NTNMR).

The initial central frequency is set based on the dashboard channel selection (F1, F2, F3 or F4) and can be manually changed either by typing in new value before or after starting the tuning process or right-click to select the following popup menu to reset the value only after starting the tuning process.
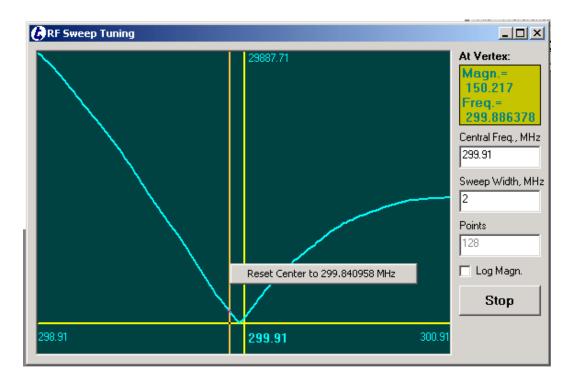
Figure 11.27 RF sweep Tuning window with center reset cursor

The initial sweep width is set with a sweep width limitation saved in the RFControl.ini file and can be manually changed by typing in a new value less than the limited sweep width before or after starting the tuning process.

The initial number of points is set based on the Points 2D value of sweep.tnt file and can be manually changed only before starting the tuning process.

The At Vertex display area is only for reference purpose to show the magnitude and frequency values of the lowest point of tuning curve displayed in this window after starting the process.

Log Magn. check box is used to select y-axis scaling mode and can be set at any time.
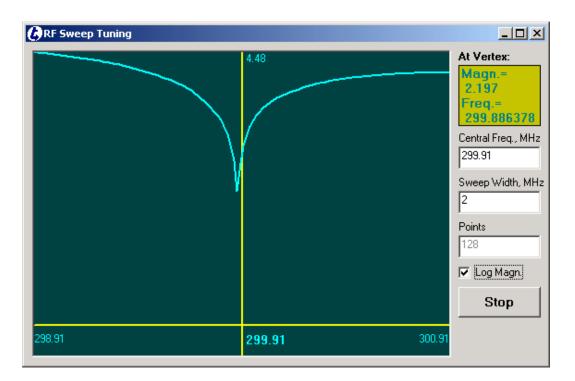
Figure 11.28 RF sweep Tuning window with Log Magn on

Simply adjust the probe to make the lowest point of the curve as close as possible to the "origin" point.

The tuning curve will only be displayed during the process. The initial x-axis width is the same as sweep width and can be changed by hitting left and right arrow keys on keyboard (sweep width will keep same by doing this.). The initial y-axis maximum is the same as magnitude maximum value and can be changed by hitting up and down arrow keys on the keyboard. The Home key on keyboard can be used to restore the full-scale of curve that uses the sweep width as x-axis width and maximum of magnitude as y-axis maximum value. However, the arrow keys and home key are only activated during the tuning process and the picture box has focus.

11.3 Tecmag Lock

11.3

11.3.1 Introduction

The Tecmag Lock Program is used to control Apollo NMR Field-Frequency Lock System. To start the program, go to Start, Program and Tecmag group, or

choose **TecmagLock** in the NTNMR Script menu. Associated file needed for this program is initialization file (lock.ini).

11.3.2 Menu Bar

Figure 11.29 Digital Lock Control menu bar

The Menu Bar includes a **Preferences** menu, **Controls** menu, and **Commands** menu, in addition to the standard File and Help menus.

11.2.3 File Menu

Figure 11.30 Digital Lock file drop down menu

The **File** menu allows the user to load and save preferences (settings or configuration) in files. The default file is ntnmr\config\TecmagLock.ini.

Open Settings…	Opens a settings (preferences) file.
Save Settings	Saves all settings in the default preferences file (ntnmr\config\TecmagLock.ini).
Save Settings As…	Saves all settings in a user-specified preferences file.
Quit (no save)	Quit the program without saving the current settings.
Exit (save)	Exit the program and save the current settings in the default preferences file (ntnmr\config\TecmagLock.ini).

11.2.4 Preferences Menu
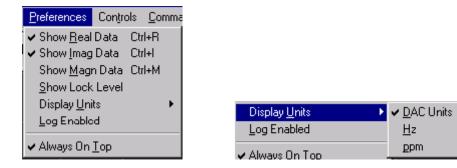
Figure 11.31 Digital Lock Preferences drop down menu

The **Preferences** menu allows the user to adjust the display and other program settings to suit the operator's preferences. The menu choices are 'toggles': when the check mark appears, the option is enabled.

Show Real Data	Displays the 'Real' part of the complex lock signal in red on the sweep display. This is the 'lock level' when the system is locked (Also Ctrl-R).
Show Imag Data	Displays the ''Imaginary' part of the complex lock signal in red on the sweep display. This is the 'error' when the system is locked. (Also Ctrl-I).
Show Magn Data	Displays the 'Magnitude' of the complex lock signal in red on the sweep display. (Also Ctrl-M).
Show Lock Level	Displays the lock level 'meter' (see below).
Display Units	Allows selection of units (DAC Units, Hz, or ppm) for the horizontal sweep display (see below). Note that the Hz and ppm values are computed, since the field is actually the swept value.
Log Enabled	Turns on logging. The lock level and error is written to a file (default ntnmr\lock.log) in the format <timestamp, level, error>.
Always on Top	Prevents the program windows from being hidden by other windows.

11.3.5 Controls Menu

Figure 11.32 Digital Lock Controls drop down menu

The **Controls** menu is used to activate any one of the four control windows (see below for details):

Display (Amplitude/Offset)	See below.
Sweep (Width/Speed)	See below.
Loop Tuning Parameters	See below.
NMR Controls	See below.

11.3.6 Commands Menu

Figure 11.33 Digital Lock Commands drop down menu

The **Commands** menu is used to execute immediate operations.

Pause Display	Pauses the display. The display is not updated, but the lock controller continues to operate normally. This is used to improve the response of the host computer..
Reset Entire Apollo System	Performs a 'Reset' on the Apollo. Equivalent to the 'hammer' on the NTNMR Acquisition toolbar.
Reset Lock Hardware	Performs a reset on the lock modules only.
Reload Lock Params	Refreshes the lock parameter values in the lock hardware modules. Useful after a 'Reset' is performed in NTNMR.

11.3.7 Lock Mode Selection Toolbar Buttons
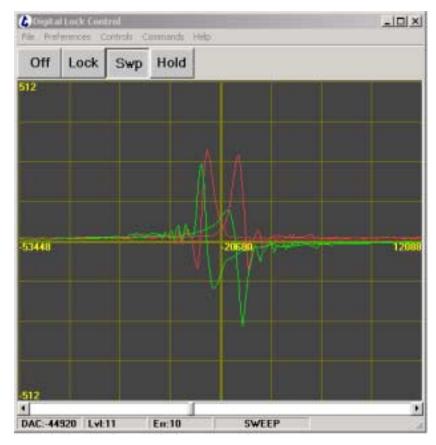
Figure 11.34 Digital Lock mode selection toolbar buttons

Clicking and selecting one of the buttons will de-select the rest. Click on Swp to start to sweep the magnetic field over the range selected in the Sweep Control window. The Lock button is for holding the resonant frequency at a certain field. Click on Hold to stop sweeping but keep pulsing.

Off	Stops the sweep, and turns off the lock transmit RF. The DAC is set to the last valid value (from Lock or from Hold mode).
Lock	Turns on the lock transmitter and receiver, and closes the control loop. The display becomes a 'strip chart'
Swp	Turns on the lock transmitter and receiver, and begins sweeping the field.
Hold	Same as Lock, except the control loop is disabled and the DAC is set to the last valid value. The value can be changed manually in this mode by right-clicking the sweep display.

11.3.8 Lock Sweep Display
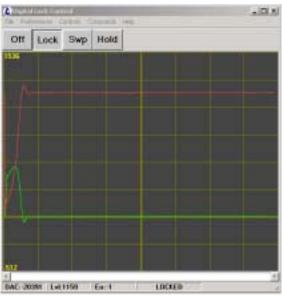
Figure 11.35 Digital Lock sweep dislpay

The ^{2}H resonance of the deuterated solvent is used to lock the NMR signal. To display the sweeping NMR signal (shown in figure above, left) press the Sweep-Button (Swp), to show the lock level (shown in figure above, right) press the Lock-Button (Lock). The error level is displayed as a green line. Using the horizontal scroll bar the center of sweep range can be shifted. Right clicking in the display will show a popup menu for setting the center of the sweep.

In the status bar (bottom of window) the current numerical values for the DAC, the lock level (forms red line) and the error (forms green line) are shown.

11.3.9 Lock Controls
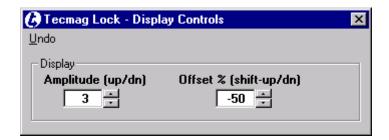
Figure 11.36 Digital Lock Display controls

- **Display Controls** is used to adjust amplitude and offset through the corresponding spin buttons (Up and down arrow keys on keyboard perform the same function).

Figure 11.37 Digital Lock Sweep controls

- **Sweep Controls** is used to select sweep width and speed though those ComboBox drop down lists.
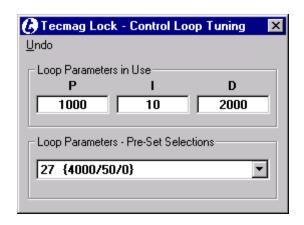
Figure 11.38 Digital Lock Loop Tuning Controls

- **Control Loop Tuning** is used to modify PID values or chose one of
  preset selections through ComboBox list.
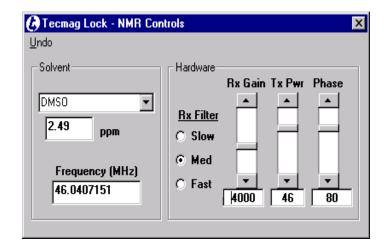
Figure 11.39 Digital Lock NMR Controls

- In **NMR Controls** window, the Solvent section is used to define the
  solvent used through the ComboBox drop down list, and corresponding
  chemical shift and frequency will be displayed in those text boxes and
  could be changed by simply typing in new values. The Hardware section
  allows users to select the receiver filter band width (Rx Filter) in one of
  the three modes, Slow, Medium, Fast, in term of the sweep rate, and to set
  the receiver gain (Rx Gain: 0 - 10000), transmitter power level (Tx Pwr: 0
  – 63.5 dB) and phase (Phase: -180° to 180°).

11.3.10 Lock Level Display

Figure 11.40 Digital Lock Level Dispaly

11.3.11 Lock Operating Procedure

1. Open the lock window. Click on the Sweep Button (Swp) to display the NMR signal. Adjust the displayed signal using the Display option (Amplitude/Offset) and the sweep rate from the Sweep option (width/speed) in the Control Menu to obtain an appropriate NMR signal display.

2. Chose the deuterated solvent for obtaining the lock signal from the NMR Control in the Control Menu. Adjust the receiver gain and the transmitter power level to obtain an appropriate size of signal. Also adjust the phase such that the displayed real part of the NMR signal) (red curve) is mostly reflecting absorption. The center frequency can be set by right-clicking on the cross point of the forward and backward sweep curve and click on Lock Button on the Lock Menu Bar.

(If no NMR signal appears in the window, increase the sweep width or adjust receiver gain and the transmitter power level.)

3. The window will now switch to the lock mode. After a short period of time for searching, the system will reach the lock status; the red line (or curve?) (lock signal) is to be above zero and the green line (error) is to be close to zero.

If the system cannot reach the lock stage, check options in step 1 and 2 with different settings.

.

11.4 Shims

11.4

To shim the magnet, two approaches are offered: Lock Shim and FID Shim. Either can be performed automatically or manually. The shim controls can be opened by clicking the Console button 🖥 in the toolbar above the NTNMR spectrum window.
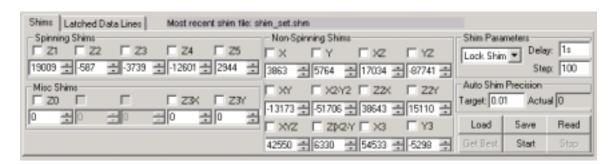
Figure 11.41 Shim window

The shim response is calculated and shown as a Shim Unit. Clicking on the NTNMR Menu toolbar View-Shim Unit, the result is shown in the upper right corner of the spectrum window.

11.4.1 Manual Shim

The shim coil current control provides the current adjustment of each individual shim coil. By editing the file ***default.shm*** (Folder "NTNMR\Shim", edit with simple text editor, e.g. Notepad) the user can customize the coil names and groups. Each coil is represented by its editable current value and a checkbox to enable it for automatic shim (checked: automatic shim mode).
The coil current can be set by manually typing the value in the respective box or by using the associated Up-Down buttons (increment is set in the Shim Parameters-Step box)

The shimming mode can be selected between FID and Lock shim mode in the Shim Parameters box. The delay between setting the shim currents and measuring the response is set with the delay value.

All shim settings are loaded or saved using the Load or Save button. Please note that the latest settings are always in the file default.shm.

Using the Read-Button the hardware shim settings can be read.

11.4.2 Auto Shim

Auto Shim maximizes the shim response using the Downhill Simplex Method described by Nelder and Mead, *Computer Journal*, **7**, (1965), 308-313. Either lock or FID responses can be selected to evaluate the homogeneity. To select the coils to be shimmed, the respective check boxes must be checked. The Auto Shim step size controls the size of the initial simplex, and hence the volume of "shim space" to be searched. In the Auto Shim Precision box, the Target value reflects the accuracy of the automated shim. Please note that big values for Target, converge fast but yield in less accurate shim, small values converge slower or never. The current value in the shimming process is shown in the Actual field. When the actual value is down to the target value the shimming stops. To start an auto shimming process, press the Start button. Clicking on the Stop button stops shimming before getting to the target.

All shim settings are loaded or saved using the Load or Save button. Please note that the latest settings are always in the file default.shm.

Using the Read-Button the hardware shim settings can be read.

11.5 NMR Data Manipulation

11.5

11.5.1 Introduction

NMR experiments often involve the determination of the relationship between two parameters. If you have a data set which looks like the following:

X	Y
1	1
2	2
…	…
N	N

You can easily conclude that the relationship between X and Y is Y = f(X) =X. Most of the time the relationship is not so obvious, and determining an expression for f(X) is not trivial. NMR data manipulation is meant to help the user find a mathematical expression that will describe his data as accurately as possible. NMR data manipulation will first display the data using linear Cartesian coordinates. NMR data manipulation will then assist the users in finding intuitive relationships to describe the data set.
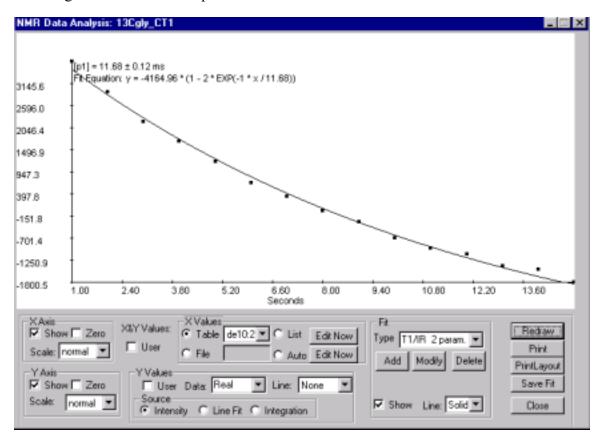
Figure 11.42 NMR Data Analysis window

Figure 11.42 shows an example of what the NMR data manipulation window looks like. The upper part of the window is a graph of both the original data and the data fit. The x-axis values of each descrete point are extracted from a selected table in the pulse sequence. The y-axis values are the cooresponding intensity values of NMR signals. The solid line represents an attempt a fitting the data with a pre-selected equation. The equation is displayed on the upper-left corner of the graph. The equation used to perform the fit can either be selected from an existing list or the user can enter his own. A number of parameters can be adjusted to improve the quality of the fit. The fitting process is controlled using setting available on the small dashboard at the bottom of the NMR data manipulation window. Keep in mind that NMR data as well as other data doesn't always follow simple mathematical expression. Often, no matter what expression is used, it will not be possible to get a perfect fit. Try to keep as many points as possible on the simulated curve and a good balance of points above and below the curve.

Controls below the graph are described in detail as follows.

- **"Y Values"** group:

Figure 11.43 "Y Values" group controls

This group is enabled when the user decides to select X data and Y data separately. When it is enabled, you can see the above controls.
  - o **"User"** Check box: if this is checked, the above controls except the check box itself will be hidden, some other controls will show instead. So let's talk about the scenario when this check box is not checked. In that case, Y data comes from NTNMR documents.
  - o **"Data"** Combo box: there are three options in this combo box. Since NTNMR data usually have real and imaginary parts, we have Real, Imaginary and Magnitude options.
  - o **"Line"** Comb box: when data are plotted as dots on the graph, user can decide to connect them by different kind of lines; or don't connect those lines at all.
  - o **"Source"** group: this is a sub-group within Y Values group. It decides whether Y data comes from Intensity, Line Fit or Integration from NTNMR document.

- **"Y Values"** group

Figure 11.44 "Y Values" group controls (with User check box checked)

When the "User" Check box in the group is checked:

- When **"User"** Check box is checked, Y data is not coming from NTNMR document. User can select their own data either from a file or create them by entering a list in a dialog box.
- **"List"** Radio button: pop up a dialog box for user to enter numbers. If user has already typed something in that dialog box, the previously typed numbers will show up for modification.
- **"Edit Now"** Button: also pop up a dialog box for user input, but if the radio button on its left is not currently selected, this option won't get activated. The list itself exists; it will be effective whenever "List" radio button is clicked.
- **"File"** Radio button: pop up File Dialog box for user to select a file as the source of Y data. But in order to make things workable, file should conform to certain format, which is quite simple. Just a bunch of numbers.

Disabled edit box on the right of "File" Radio button shows the name of file user loaded.

- **"X Values"** group:

Figure 11.45 "X Values" group controls

This group is enabled when user decides to select X data and Y data separately. When it is enabled, there are 4 options to choose from. The number of Y data in this case determines the number of X data.

- **Table** Radio button: sequence tables come from sequence.
- **File** Radio button: any file can be loaded.
- **List** Radio button: when this option is selected, a dialog box will come up for user to type numbers. If user has already typed something in that dialog box, the previously typed numbers will show up for modification.
- **Auto** Radio button: when this option is selected, a different dialog box pops up. Since the number of X data is already determined by number of

Y data, the dialog box asks user to type starting number and step. With this information, a series of numbers will be created.
- o **Edit Now** Buttons in this group work the same way as their counterpart in the "Y Values" group.

- **"X&Y Values"** checkbox:

Figure 11.46 "X & Y Values" checkbox

If this checkbox is checked, then both X and Y data should come from a particular file on disk. The format should be very simple, like this:
3   5.5
4   6.6
…

- **"X Axis"** group:
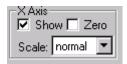
Figure 11.47 "X Axis" group controls

This group determines the way X Axis is drawn.
- o **Show** Check box: determine whether show or not show X Axis at all.
- o **Scale** Combo box: has a bunch of options to scale the axis. They are: normal, log, ln, 1/log, 1/ln, 1/normal. They are quite self-explainable.
- o **Zero** Check box: if this check box is checked, the graph will show the Zero position explicitly. See if X data spans from negative values to position values, then a vertical dash line will be drawn at the zero point as the graph below:
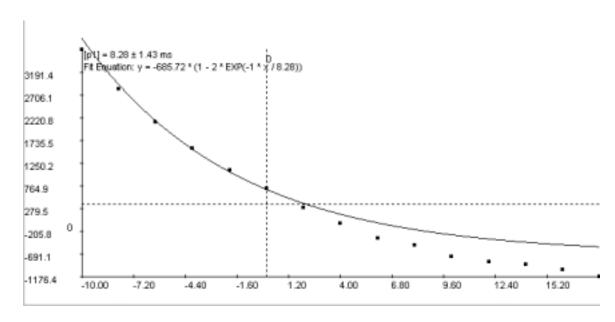
Figure 11.48 Graph of relaxation data with fit

- **"Y Axis Group"**

Figure 11.49 "Y Axis" Group control

This group determines the way Y Axis is drawn.
The controls in this group work pretty much the same way as their counterparts
in the **"X Axis"** group. Notice the horizontal dash line is drawn as an indication
of zero Y position.

- **Fit** Group:

Figure 11.50 "Fit" Group control

- o **Type** Combo box: this combo box loads all the fit equations user can
  use. The first 5 equations are built-in. User can add their own equations
  for fit purpose.
- o **Add** button: Click on this button will pop up a dialog box for the user to
  enter their own equation as well as initial values for those unknown

parameters. When the user clicks OK on the dialog box, the new equation and the initial values are loaded and the user can see the equation in the above combo box.

o **Modify** button: User can select a self-defined equation and then click on this button to modify it, either the equation itself or the initial values. Pre-defined equations can't be modified.

o **Delete** button: User can delete self-defined equation. Pre-defined equations can't be deleted.

o **K** Edit box: it is used only when the above pre-defined equation is used. Other equations won't use it at all. When the above equation is not active, this edit box won't show up.

o **Show** check box: decide whether show the fit curve or not.

o **Line** combo box: decide which type of line is used to draw the fit curve.

- **Other controls**:

Figure 11.51 Other control buttons

o **Redraw** button: click on this button will force Window to refresh the graph. For some reason Window is not always refreshed when something changed, that's the purpose of this button.

o **Print** button: print the graph.

o **PrintLayout** button: If a print layout is already opened, clicking on this button will add the graph to the print layout.

o **Save Fit** button: Save a text file which shows things like Data points, fit points, fit results, date of experiment, and so on. It is only useful when the Data source is the NTNMR document.

o **Close** button: dismiss NTNMR Data Analysis window.

11.6 NMRWizard

11.6

11.6.1 Introduction

The NMRWizard program is a simple user interface for automated operation of an NMR spectrometer. To start the program, go to Start, Program and Tecmag group, or choose **NMRWizard** in the NTNMR Script menu. The program operates in two modes: experiment (default) and setup.
Experiment mode allows any user to run one of the pre-defined experiments. An "experiment" consists of an NTNMR data file (*.tnt) that defines the pulse sequence, parameters, number of scans, etc., together with the various options defined by the authorized user via the setup mode. It may also include one or more script files (*.vbs) and print layout files (*.prt).
Setup mode allows an authorized user to define experiments.

11.6.2 Starting the Wizard

The program begins with a **Welcome Screen** (figure 11.52). The **Welcome Screen** contains the program version, a brief description of the program, copyright information as well as the following control components:

- **Setup Mode** checkbox: enable Setup Mode.
- **Exit** button: click to exit the program.
- **Start** button: click to start the chosen mode.
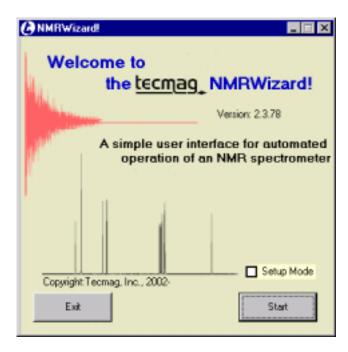
Figure 11.52 NMRWizard welcome screen

11.6.3 Experiment Mode

The title **NMRWizard!** indicates the program is in experiment mode. There are always three command buttons:

- **Exit** to quit the NMRWizard program.
- **Back** to go back to previous window (where applicable).
- **Next** to continue to the next window(step).
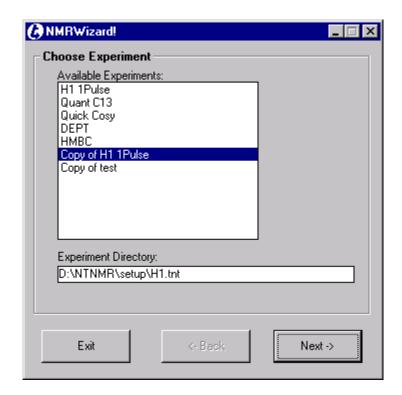
Figure 11.53 NMRWizard **Choose Experiment** window

Click one of the available experiments in the list to choose it. The path to the corresponding experiment file is shown. Click Next to continue with the next step.
The **Sample Control** window (figure 11.54) will only appear if the sample control hardware is present (defined in NMRWizard.ini). The Eject command button is also optional depending on the hardware.
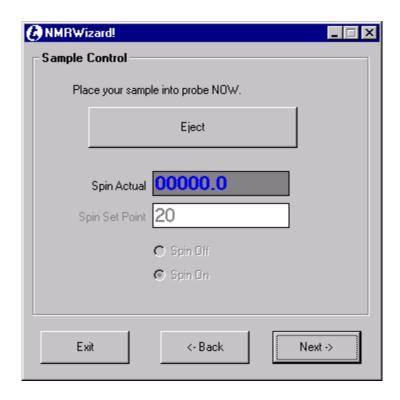
Figure 11.54 Sample Control window **eject**

Now put your sample into probe as follows: click **Eject** to turn on the eject air. The caption of the button becomes **Ejecting**. Wait until ejecting completes as indicated by the caption changing to **Insert**. Place your sample into the probe stack. Then click **Insert**. The caption becomes **Inserting**. Wait until inserting completes, indicated by the caption changing back to **Eject**. See figures 11.54, 11.55, 11.56, 11.57.
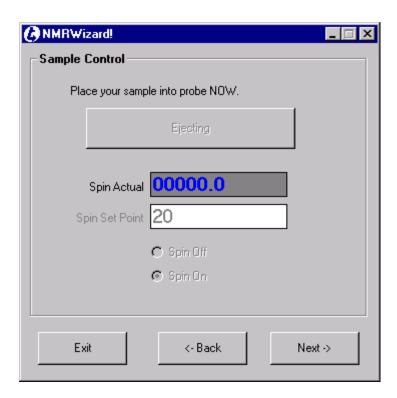
Figure 11.55 Sample Control window **ejecting**
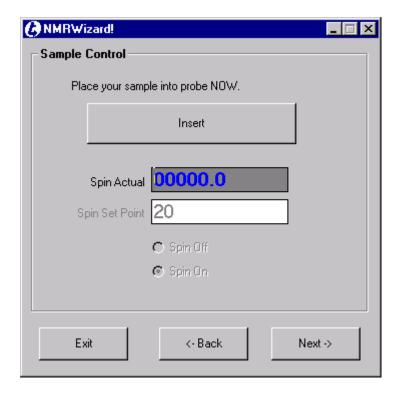
Figure 11.56 Sample Control window **insert**
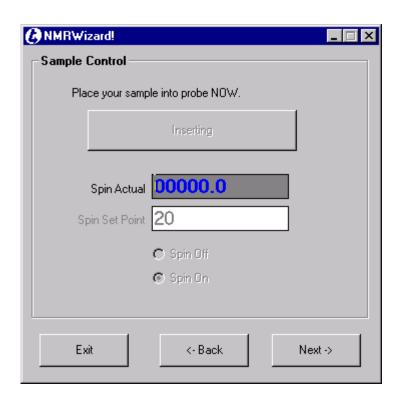
Figure 11.57 Sample Control window **inserting**

If Spin On was specified for the experiment, it is important to wait until the actual *spin rate is stabilized* before going to the next step. Click Next to continue with the next step.

The **Select Solvent** window (figure 11.57) will only appear if the field-frequency lock hardware is present (defined in NMRWizard.ini). Choose one of the deuterated lock solvents from the list. Click Next to continue with the next step.
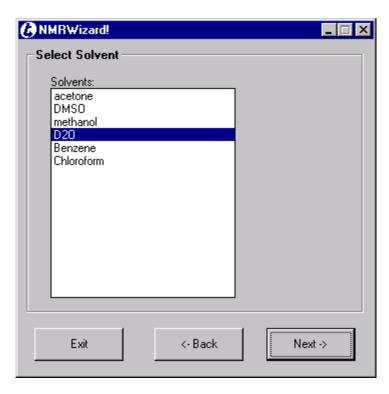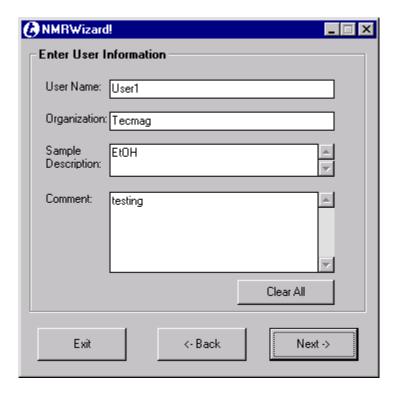
Figure 11.57 Select Solvent window

Figure 11.58 Assign Output File window:

Type a file path or click the browse button (**…**) to specify an output file to save your data. Click Next to continue with the next step.

Figure 11.59 Enter User Information window:

Input all necessary user information here or click **Clear All** button to clear all fields. User information will be saved into the NMR data file's comment field after the experiment is run. Click Next to continue with the next step.

Figure 11.60 Experiment Summary window: Idle
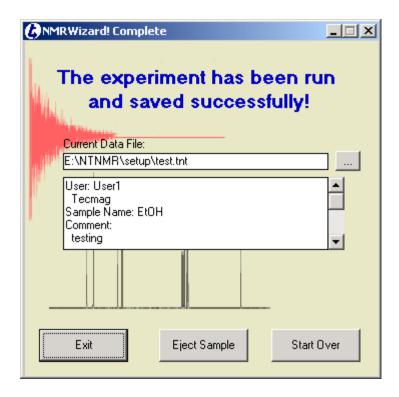
All the experiment's information (including the user information) is displayed in the Experiment summary window. This experiment summary will be saved into the NMR data file's comment field after the experiment is run. Click **Run Exp.** to begin the experiment. The progress bar and status text field show the on-going experiment status. The caption of Run Exp. button changes to "Stop Exp." and both **Exit** and **Back** are disabled. The experiment may be stopped any time by clicking the Stop Exp. button, in which case the status text shows **Early Stop** and the button caption changes back to **Run Exp**. See the figures 11.60, 11.61, 11.62.

Figure 11.61 Experiment Summary window: in progress

Figure 11.62 Experiment Summary window: Early Stop

Figure 11.63 Experiment Complete window:

When the experiment completes, the program goes to the **Experiment Complete** window. The window contains information about the current experiment while NTNMR shows the actual data file. The browse button (…) offers the ability to examine any other NMR data files.

There may be three command buttons:
- **Exit** to quit the program;
- **Start Over** to go back the Welcome screen;
- **Eject Sample** button (if applicable) to change or remove the sample.

11.6.4 Setup Mode

All Experiments must be defined via Setup mode before they can be run. Setup mode is password protected and the password is defined in the NMRWizard.ini file. The Setup Mode Login box allows the user to input the password. The password is case sensitive. Click **OK** to continue or **Cancel** to go back to the Welcome screen.
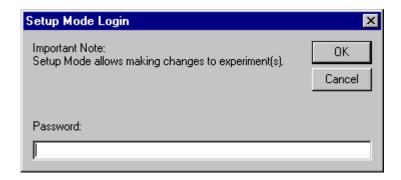
Figure 11.64 Setup Mode Login window

The title bar indicates **Setup Mode** during the entire setup process until you logout. There are always three command buttons:
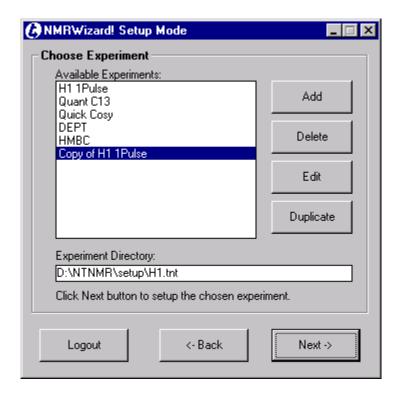
- **Logout** to quit Setup Mode and return to the Welcome screen;
- **Back** to go back to previous window, it is disabled for the first window(step); **Next** to continue next step.

Figure 11.65 Hardware preference window

Check the Sample Eject/Insert Device if it is required. For instance, it may be unchecked for a solid probe or checked for a liquid probe. (If no sample control

hardware exists (defined in NMRWizard.ini file), the checkbox will be disabled.)

Figure 11.66 Choose Experiment window:

All the existing experiments are listed in the Choose Experiment window. An experiment can be selected by clicking on it and the experiment file location will be displayed in the Experiment Directory: dialog box. The user can add, delete, edit or duplicate an experiment by clicking one of the four command buttons as follows:

- **Add**: popup a dialog box to allow the user to select an existing experiment file, followed by an input box to allow user to type a name for the new experiment. Then the new experiment will be added to the list. Once an experiment has been created, click **Next** to setup the new experiment.
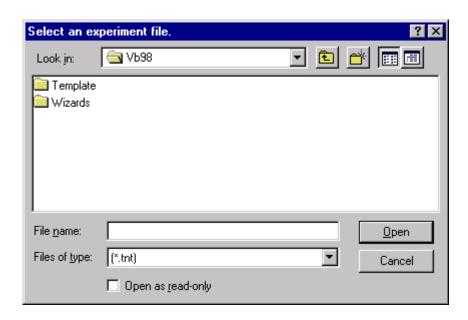
Figure 11.67 Select an experiment file dialog window
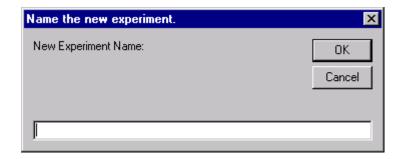
Figure 11.68 Name the new experiment dialog window

- **Delete**: popup a message box to confirm the action. If yes, the selected experiment will be deleted from the list permanently.

Figure 11.69 experiment deletion dialog window

- **Edit**: only used to change the experiment file and/or experiment title. The configuration of any selected experiment can be edited/changed by clicking **Next** button.

- **Duplicate:** copy the selected experiment configuration and add it to the list with "Copy of " before the selected name. The name can then be changed by the **Edit** command and the configuration can be changed by the **Next** command.
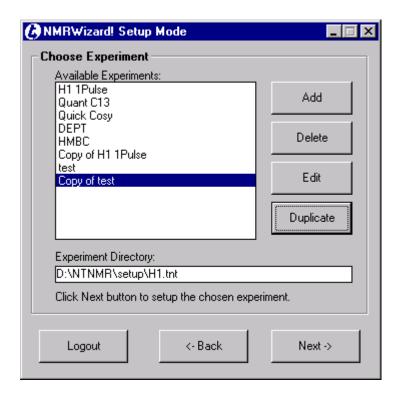
Figure 11.70 NMRWizard Setup window, Duplicate mode

The Sample Control window (Figure 11.71) will only appear if the sample control hardware is present (defined in NMRWizard.ini). This window defines the sample spin state (on/off) and set point.
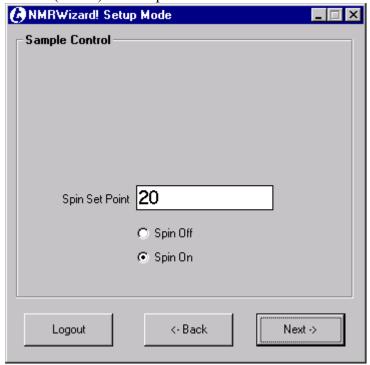
Figure 11.71 Sample Control window:

For each experiment you can use one or more of the built-in setup options, or control the setup via a script file (user supplied).
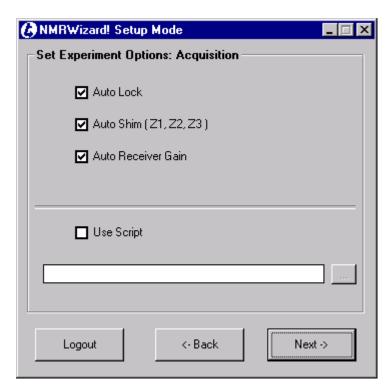
Figure 11.72 Set Experiment acquisition options window

Check all applicable functions. **Auto Lock** and **Auto Shim** may be disabled and unchecked if lock module and/or shim module do not exist (defined in NMRWizard.ini file). If **Use Script** is checked the **Run Script Prior To Above** (figure 11.73) will be visible and this will allow user to decide to run script file first or not. Checking **Use Script** will also enable the browse button (…), which is used to locate an existing script file.
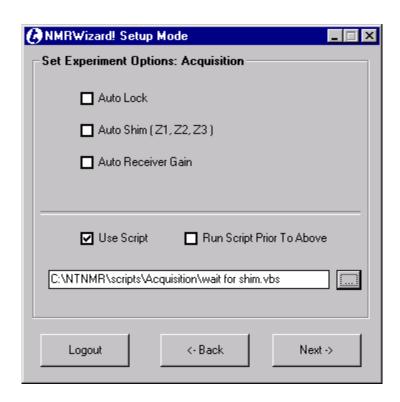
Figure 11.73 Set Experiment acquisition options window:script selection
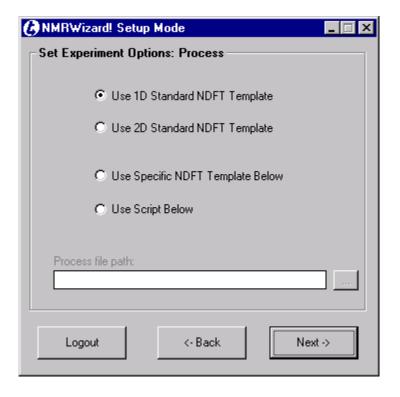
Figure 11.74 Set Experiment Options: Process window

Select one of the options for processing data:

- **Use 1D Standard NDFT Template**: The standard tft file must be present, which typically includes BC, EM, FT, and PC. The file can be modified to meet certain requirements.
- **Use 2D Standard NDFT Template**: The standard tft file must be present, which typically includes 2nd dimension complex transpose, 1st dimension BC and FT, and 2nd dimension BC and FT. The file can be modified to meet certain requirements.
- **Use Specific NDFT Template Below**: browse button (…) is enabled to locate an existing NDFT template file (*.tft).
- **Use Script Below:** browse button (…) is enabled to locate a user-defined script file (*.vbs).
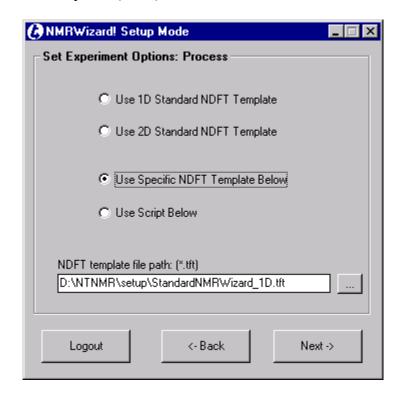
Figure 11.75 Set Experiment Options: Process template window
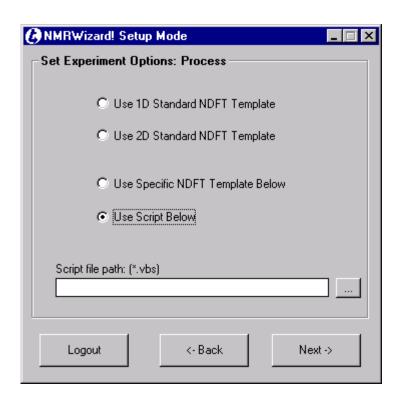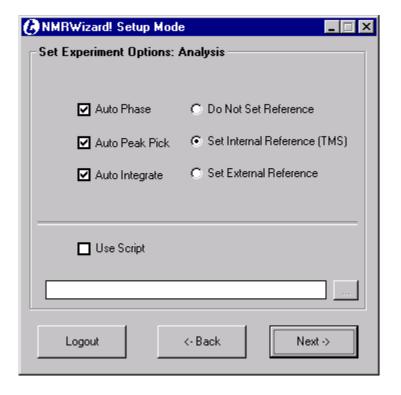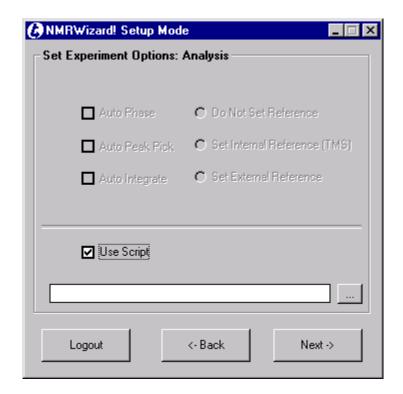
Figure 11.76 Set Experiment Options: Process script window
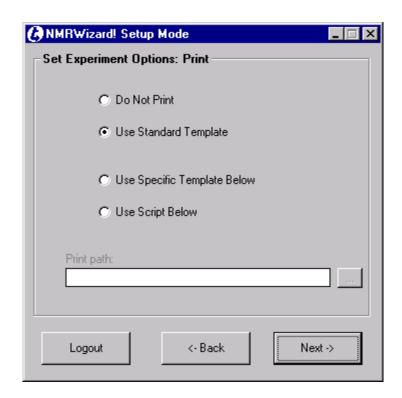
Figure 11.77 Set Experiment Options: Analysis window:

Check and select all applicable functions. The reference can be set internally with TMS or externally with *pre-defined* water frequency in D2O inside the lock.ini file. If **Use Script** is checked all others will be unchecked and disabled; uncheck **Use Script** will enable all others and reset them to previous status. Checking **Use Script** will also enable the browse button (…), which is used to locate an existing script file for the data analysis.
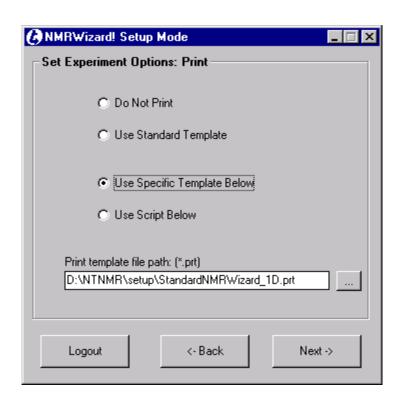
Figure 11.78 Set Experiment Options: Analysis script window:

Figure 11.79 Set Experiment Options: Print window:

Select one of the options for printing data: using the standard template (the standard prt and par files must be present); using a specific template (browse to locate the existing prt file while browse button (…) is enabled); or using a user-defined script (browse to locate the existing vbs file while browse button (…) is enabled).

Figure 11.80 Set Experiment Options: Print from template window:
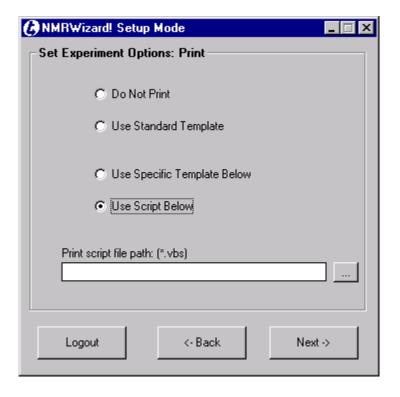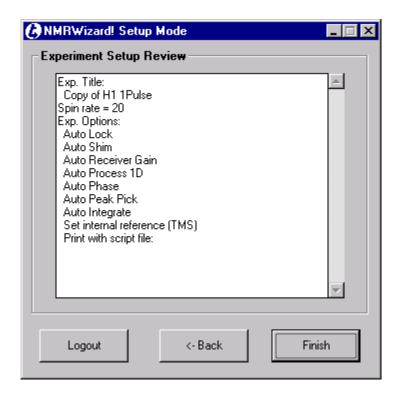
Figure 11.81 Set Experiment Options: Print frm script window:

Figure 11.82 Experiment Setup Review window:

Review the configuration of the experiment. Click **Finish** to save the experiment configuration and go on to the next screen. The experiment setup information is stored in the NMRWizard's own configuration file (NMRWizard.INI).
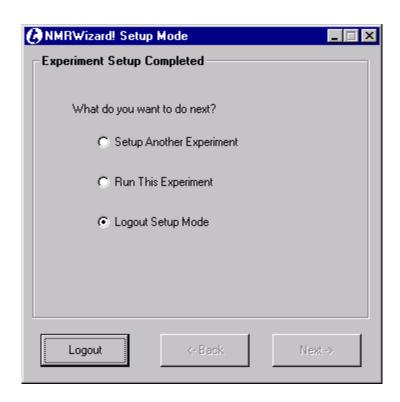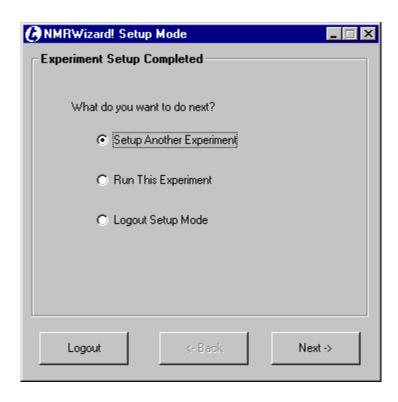
Figure 11.83 Experiment Setup Completed window: logout option

This is the final step of Setup Mode. By default **Logout Setup Mode** option is always selected and both **Back** and **Next** buttons are disabled. Click **Logout** button to return to the Welcome screen. Chose the other two options will enable **Next** button. **Setup Another Experiment** will go to Choose Experiment window and stay logged in Setup Mode; Run This Experiment will logout Setup Mode and enter Experiment Mode.

Figure 11.83 Experiment Setup Completed window: setup another experiment option

Figure 11.83 Experiment Setup Completed window: Run Option

11.6.5 Files requires for NTNMRWizard

Associated files needed for this program are:
- Initialization files: (:\ntnmr\config\)
  NMRWizard.ini,
  lock.ini;
- Standard NDFT template files: (:\ntnmr\setup\)
  StandardNMRWizard_1D.tft
  StandardNMRWizard_2D.tft
- Standard print template files: (:\ntnmr\setup\)
  StandardNMRWizard_1D.prt,
  StandardNMRWizard_1D.par,
  StandardNMRWizard_2D.prt,
  StandardNMRWizard_2D.par;
- Various experiment files (*.tnt);
- TecmagLock program if applicable;
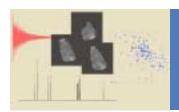
SampleInterface program if applicable.

12 - Automation

12.1 Introduction

12.1

NTNMR provides the ability to automate many of its processes. This allows sophisticated acquisition and processing routines to be created that can be executed from a single click. Applications include complex data processing, automatic execution of a suite of experiments, calibration experiments or even creating a specialized virtual replacement for the NTNMR interface.

The term "Scripting" has been adopted to refer to automating processes in NTNMR because the capabilities far surpass what are typically referred to as "Macros." Conceptually, however, "scripts" in NTNMR can be thought of as "macros."

12.2 OLE Automation

12.2

OLE (Object Linking and Embedding) is a method created by Microsoft to allow one application to be controlled by another. NTNMR supports OLE Automation (but *not* Linking and Embedding at this time). This method allows the "Objects" in NTNMR to be "exposed" so that another application can control it. Microsoft Word and Microsoft Excel are other applications that are "OLE compliant."

Both the *Document* object and the *Application* object are exposed in NTNMR. Many commands can be accessed through either object. However, some care must be used in selecting the object, as unexpected results can occur if the active document (generally, the front most 1D display window) changes during execution of a script.

Other programs are also part of the NTNMR package, for example TecmagLock, RF Control, Autoshim, or SampleInterface. Each of these programs has its own set of automation commands, mostly accessible through the respective *Application* objects.

It is useful to familiarize yourself with the Microsoft terminology uses when referring to OLE. The *OLE Server* is the application that exposes its objects to other applications - any of those objects can be used in a Visual Basic application. The *OLE Client* is the application that controls the *OLE Server*. The client can be a 'stand alone' application written in Visual Basic (or another OLE Compliant programming language).

Further Information about OLE automation is available from a variety of books and Internet web pages. Any further discussion of OLE technology is beyond the scope of the manual, and furthermore, is not necessary for writing "scripts" to control NTNMR.

12.3

12.3 Visual Basic & VBScript

12.3.1 General Information

Visual Basic 6.0 and VBScript have been adopted by Tecmag as the "official supported" scripting languages. Many other languages such as WinBatch, Visual C++, Java and Python have the ability to issue OLE commands and can be used to control NTNMR. The discussion in this manual refers specifically to Visual Basic 6.0 and VBScript; although most of these concepts should generally apply to any other language that may be used to control NTNMR.

12.3.2 Visual Basic or VBScript?

As a general rule, the more complicated the NMRScript, the more likely it is that one would want to use Visual Basic, instead of VBScript. It is important to note that Visual Basic is a programming language, with its own development environment, while VBScript is very much a scripting language. VBScript is provided with NTNMR, while Visual Basic requires the user to buy the software development package from Microsoft.

12.3.3 Compatibility and Support Note

While it is possible to use other programming languages to control NTNMR, Tecmag is unable to guarantee performance and compatibility with other languages. Furthermore, the degree to which Tecmag may be able to support "scripting" languages other than Visual Basic 6.0 may be limited.

12.4 The Script Menu

12.4
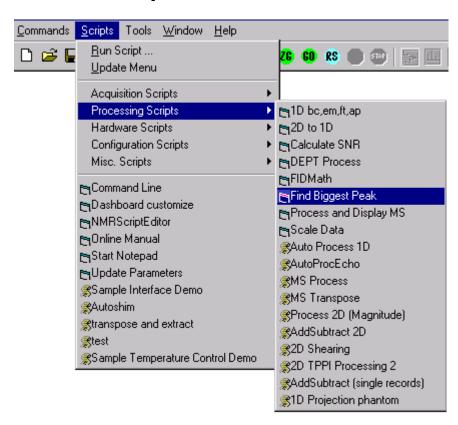
Figure 12.1 The script pull down menu

The **Script** menu in NTNMR allows the user to create, manage, and execute scripts saved as an executable (*.exe) or NMRScript (*.vbs) files, without leaving NTNMR. All scripts saved in **NTNMR\scripts** will appear in the menu in NTNMR. There are also 5 pre-defined subdirectories (**Acquisition**\, **Processing**\, **Hardware**\, **Configuration**\, **Misc.**\) whose contents will appear in the corresponding sub-menus in NTNMR. Note that *any* executable file that is saved in these script folders will appear in the Script Menu.

Run Script	Locate and run a script located anywhere on disk.
Update Menu	Refresh the menus if a script is added while NTNMR is already running.
Acquisition Scripts	Open the Acquisition Scripts submenu (scripts located in \NTNMR\scripts\Acquisition\ ).
Processing Scripts	Open the Processing Scripts submenu (scripts located in \NTNMR\scripts\Acquisition\ ).
Hardware Scripts	Open the Hardware Scripts submenu (scripts located in \NTNMR\scripts\ Processing\ ).
Configuration Scripts	Open the Configuration Scripts submenu (scripts located in \NTNMR\scripts\ Configuration\ ).
Misc. Scripts	Open the Acquisition Scripts submenu (scripts located in \NTNMR\scripts\ Misc.\ ).
. . . . .	Remaining entries are the names of actual scripts which can be selected directly from the menu (scripts located in \NTNMR\scripts\ ).

As seen in the figure, the menu uses two different icons to indicate the type of script:

The "scroll" icon indicates an NMRScript file (*.vbs); while the "window" icon indicates a compiled, executable file (*.exe).

12.5 Automation Commands - Abbreviated List

12.5

The following tables contain shorthand listings of all commands that are currently scriptable in NTNMR and its accessory programs. Commands in this list are organized by functionality. Readers may wish to refer to the alphabetical list in section 10.6 of this manual for specific details of each command. Those commands that are specified as Automation Only, are not available through the NTNMR graphical user interface. Refer to the next section for specific examples of each command. Refer to Microsoft documentation for Windows scriptable objects, such as the FileSystem.

Scriptable Command: Name(s) of command as used in Visual Basic.
AO: Command is available only through automation, i.e. Automation Only

Parameters, name: Arguments to be supplied with command

Parameters, Type: Type of data to be supplied or returned
Return Type: Type of the function return value (if any)

Dashboard: If the command uses parameters from the dashboard, they are listed

Note: Most of NTNMR commands are *both document and application* object commands. These commands appear in the first list. Commands specific to the *application* object appear in the second list. Commands for accessory programs (see Chapter 11) are in subsequent lists.

NTNMR.Application/Document Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
Acquisition Commands					
ZeroAndGo, ZG	no	-	-	-	-
Go	no	-	-	-	-
Stop	no	-	-	true/false	-
Abort	no	-	-	true/false	-
Reset	no	-	-	true/false	-
CheckAcquisition	yes	-	-	variant	-
Processing Commands - Baseline					
BaselineCorrection, BC	no	-	-	-	-
BaselineTilt, BT	no	-		-	-
Processing Commands - Apodization					
ExponentialMultiplication, EM	no	[optional] lb	variant	-	LB 1D
GaussianMultiplication, GM	no	[optional] gb	variant	-	GB 1D
DoubleExponentialMult, DM	no	[optional] db	variant	-	DM 1D
SinBell, SB	no	[optional] shift	double	-	SB Shift 1D
		[optional] width	double		SB Width 1D
		[optional] skew	double		SB Skew 1D
SinBellSquared, SBS	no	[optional] shift	double	-	SB Shift 1D
		[optional] width	double		SB Width 1D
		[optional] skew	double		SB Skew 1D

Trapezoid, TZ	no	[optional] tz1	long	-	TZ 1 1D
		[optional] tz2	long		TZ 2 1D
		[optional] tz3	long		TZ 3 1D
		[optional] tz4	long		TZ 4 1D
TrafWindow, TW	no	[optional] tw	double	-	Traf 1D
BlackmanHarris, BH	no	[optional] shift	variant	-	SB Shift 1D
		[optional] width	variant		SB Width 1D
EchoExponentialMult, EEM	no	[optional] lb	variant	-	LB 1D
EchoGaussianMult, EGM	no	[optional] gb	variant	-	LB 1D

Processing Commands - Transforms

| FourierTransform, FT | no | - | | - | - |
| InverseFT, IFT | no | - | | - | - |

NTNMR.Application/Document Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		

Processing Commands - Transforms (cont.)

RealFT, RFT	no	-		-	-
EchoFT, EFT	no	-		-	-
HilbertTransform, HT	no	-	-	-	-

Processing Commands - Phasing

PhaseCorrection, PC	no	[optional] ph0	variant	-	Sys. Phase 0 1D, 2D 3D, 4D
		[optional] ph1	variant		Sys. Phase 1 1D, 2D 3D, 4D
AutoPhase, AP	no	-	-	-	-

Processing Commands – ND Processing

NDFT	no	[optional] template	variant	-	-
SetNDPhase	no	Dim	short	true/false	-
		ph0	float		
		ph1	float		
SetNDEchoCenter	no	dim	short		-
		center	long		
SetNDApodization	no	dim	short	true/false	-
		par1	float		
		par2	float		
		par3	float		
		par4	float		

SetNDTranspose	no	dim	short,	true/false	-
		type	short,		
		data	short		
SetNDParameters	no	dim	short	true/false	-
		BC	true/false		
		ZeroFill	short		
		FT	short		
		Reverse	true/false		
		Phase,	short		
		Apodization	short		

NTNMR.Application/Document Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
Processing Commands – ND Processing (cont.)					
NDTranspose	yes	nX	long	true/false	-
		nY	long		
		nZ	long		
		nD	long		
NDComplexTranspose	yes	nX	long	true/false	-
		nY	long		
		nZ	long		
		nD	long		
NDFTLoadTemplate	no	file	string	long	-
Processing Commands – Data Organization					
ZeroFill, ZF	no	[optional] points	long	-	-
EchoZeroFill, EZF	no	[optional] points	variant	-	-
ReadFirstHalf	no	-	-	-	-
ReadSecondHalf	no	-	-	-	-
LeftShift	no	[optional] points	variant	-	Shift # points
		[optional] zero out	variant		
RightShift	no	[optional] points	variant	-	Shift # points
		[optional] zero out	variant		
GetDataSize	yes	-	-	long	-
GetNDSize	yes	dimension	long	long	-
ReSetDimensionInfo	yes	n1D	long	true/false	-
		n2D	long		
		n3D	long		
		n4D	long		

Transpose	no	-		-	-	-
ReverseSpectrumRows	yes	-		-	-	-
ReverseSpectrumColumns	yes	-		-	-	-
SortDataPoints	no	pattern		string	-	-
DigitalCorrect	no	-		-	-	-

Processing Commands – Data Point Manipulation						
GetData	yes	-		(array)	variant	-
SetData	yes	nPointNumber		long	true/false	-
		cplxPoint		variant		

NTNMR.Application/Document Object Commands

Scriptable Command	AO	Parameters		Return	Dashboard
		Name	Type		
Processing Commands – Data Point Manipulation					
GetDataPoint	yes	nPointNumber	long (array)	variant	-
SetDataPoint	yes	point, array	long, array	true/false	-
Get1Drecord	yes	record	long	variant	-
Get1Dsize	no	-	-	long	-
GetReal	no	-		variant	-
ZeroImaginary		nRecord	long	true/false	
File Commands					
Save	no	-	-	-	-
SaveAs	no	file path	string	-	-
OpenFile	no	file name	string	true/false	-
LoadSequence	no	file path	string	true/false	-
SaveSequence	no	-	-	true/false	
SaveSequenceAs	no	file path	string	true/false	-
GetSequenceName	yes	file path	string	string	-
GetActiveDocPath	yes	-	string	string	-
Export	no	file path	string	true/false	-
		type	long		
SaveAsJPG	no	file path	string	-	-
		width	long		
		height	long		
AppendToFile	no	filename	string	true/false	
		new file	long		

ExtractRecord	no	Index2D	long	true/false	-
		Index3D	long		
		Index4D	long		
		filename	string		

Display Commands

FitToWindow	no	-	-	-	-
AutoDisplayWhileProcessing	yes	-	-	true/false	-
UpdateDisplay	yes	-	-		-
SetNDRecordPointer	no	dimension	long	true/false	-
		record	long		

NTNMR.Application/Document Object Commands

Scriptable Commands	AO	Parameters		Return	Dashboard
		Name	Type		

Display Commands(cont.)

SetRecordPointer	no	n2D	long	true/false	-
		n3D	long		
		n4D	long		
GetNDRecordPointer	yes	dimension	long	long	-
SetSelection	yes	start	long	true/false	-
		end	long		
SetCursorPosition	yes	cursor position	long	true/false	-
GetCursorPosition	yes	-	-	long	-
Get1Dselection	yes	start point	string	string	-
		end point	string		
Get1DselectionEnd	yes	end point	long	long	-
Zoom	no	nStart	long	true/false	-
		nEnd	long		

2D Display Commands

OpenTwoDView	-	[optional] data type	variant	true/false	-
		[optional]display mode	variant		
		[optional] aspect	variant		
Set2Dcursor	yes	nAxis	integer	true/false	-
		nPoint	long		
Get2Dcursor	yes	nAxis	integer	long	-
Set2DNumPanes	no	number	integer	true/false	

Sequence Editor Commands					
Compile		-	-	true/false	
SetTable	yes	table name	string	true/false	-
		entries	string		
GetTable	yes	table name	string	string	-
GetTableList	yes	-	-	string	-
Miscellaneous Commands					
SetNMRParameter	yes	name	character	-	-
		value	variable		
GetNMRParameter	yes	name	string	variant	-
GetComment	yes	-		string	-

NTNMR.Application/Document Object Commands					
Scriptable Commands	**AO**	**Parameters**		**Return**	**Dahsboard**
		Name	**Type**		
Miscellaneous Commands(cont.)					
SetComment	yes	comment	string	-	-
TextToTime	yes	time text	string	double	-
Option Mode Commands – Baseline Fix					
AutoBaselineFix	no	function type	string	true/false	-
		num of points	long		
		apply to all	true/false		
DoBaselineFix	no	function type	string	true/false	-
		num of points	long		
		array list	string		
		units	string		
		apply to all	true/false		
Option Mode Commands – Peak Pick					
AutoPeakPick	no	-	-	string	-
ClearPeakPick	no	-		-	-
GetPeakPickList	yes	units	string	string	-
GetPeakPickAmplitudes	yes	-		string	-
PickOnePeak	no	position	double	-	-
		units	string		
ShowPeakPicks	no	show	true/false	-	
SavePeakPickList	no	filename	string	-	-

Option Mode Commands – Line Fits					
AddOneLineFit	no	left	double	-	-
		right	double		
		units	string		
ShowLineFits	no	show	true/false	-	-
Option Mode Commands – Integration					
AutoIntegrate, AI	no	position	double	-	-
		units	string		
ClearIntegrals	no		-	-	-

NTNMR.Application/Document Object Commands					
Scriptable Commands	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
Option Mode Commands – Integration(cont.)					
Integrate	no	left	double	double	-
		right	double		
		parameters	string		
LoadIntegrals	no	file path	string	true/false	-
SaveIntegrals		start	double	true/false	
		end	double		
		units	string		
		file path	string		
SaveAllIntegrals	no	filename	string	true/false	-
SaveIntegralsAsText	no	file path	string	true/false	-
ShowIntegrals	no	show	true/false	-	-
					-

NTNMR.Application Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return Type**	**Dashboard**
		Name	**Type**		
File Commands					
CloseActiveFile	no	-	-	true/false	-
CloseFile	no	filename	string	true/false	-
Print Commands					
AddActiveDocToPrintPreview	yes	table name, entries	string	true/false	-
ClosePrintPreview	no	-	-	true/false	
LoadPrintPreviewTemplate	no	filename	string	true/false	
PrintFromPreview	no	-	-	true/false	
Miscellaneous Commands					
GetDocumentList	no	-	string	-	
HideWindow	no	-	-	-	-
ShowWindow	no	-	-	-	-
Quit	no	-	-	-	-
Dashboard Commands					
GetParameterPageList	yes	-	string	string	-
GetParameterListInPage	yes	page name	string	string	-
SetParameterListInPage	yes	page, parameter list	string	-	-
AddParametersToPage	yes	page, parameter list	string	-	-
RemoveParameterPage	yes	page	string	-	-
RemoveAllPages	yes	-	-	-	-
LoadParameterSetupFromFile	no	path name	string	true/false	-
SaveParameterSetupToFile	no	path name	string	true/false	-
SetActiveParameterPage	no	page	string	true/false	-
LoadParametersFromFile	no	path name	string	true/false	-
SaveParametersToFile	no	path name	string	true/false	-
Hardware Commands					
SoftStop	no	-	-	long	-
RepeatScan	no	-	-	true/false	-
SetLatchedData	no	data	long		true/false
GetLatchedData	no	-	-	long	-

NTNMR.Application Object Commands					
Scriptable Commands	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
Shim Commands					
CountShims	?	-	-	long	-
ExistShim	no	which shim	string	true/false	
GetOneShim	?	which shim	string	long	-
SetOneShim	no	shim	string	true/false	-
		value	long		
ActivateShims	no	list of shims	string	true/false	-
StartShims	no	-	-	-	-
GetShims	no	-	-	variant	-
CheckShimProgress	yes	false = complete	-	true/false	-
GetActualShimPrecision	no	-	-	double	-
SetAutoShimParameters	no	delay	string	-	-
		precision	float		
		type	integer		

TecmagLock Application Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
ShowLockWindow	yes	-	-	-	-
LockSweepSpeed	no	-	-	long	-
LockSweepSpeed	no	-	long	-	-
LockSweepWidth	no	-	-	long	-
LockSweepWidth	no	-	long	-	-
LockFieldCenter	no	-	-	single	-
LockFieldCenter	no	-	single	-	-
LockSolventPPM	no	-	-	single	-
LockSolventPPM	no	-	single	-	-
LockSolventText	no	-	-	string	-
LockSolventText	no	-	string	-	-
LockTxPower	no	-	-	single	-
LockTxPower	no	-	single	-	-
LockRxGain	no	-	-	single	-
LockRxGain	no	-	single	-	-
LockNMRPhase	no	-	-	single	-
LockNMRPhase	no	-	single	-	-

TecmagLock Application Object Commands					
Scriptable Commands	**AO**	**Parameters**		**Return**	**Dashboard**
		Name	**Type**		
GetLockState	yes	-	-	integer	-
LockLevel	yes	-	-	single	-
LockLoopChar	yes	-	integer	-	-
StartLock	no	-	-	-	-
StartAutoPhase	no	-	-	-	-
StartAutoLock	no	-	-	-	-
StartHold	no	-	-	-	-
StartSweep	no	-	-	-	-
StopLock	no	-	-	-	-

SampleInterface Application Object Commands					
Scriptable Command	**AO**	**Parameters**		**Return Type**	**Dashboard**
		Name	**Type**		
SpinRate	no	-	-	Double	-
SpinRateSetPoint	no	-	Double		-
SpinSample	no	0, 1, 2 0=Spin off 1=Spin with close loop 2=spin with open loop	-	-	-
SampleTemperature	no	-	-	Double	-
VTSetPoint	no	-	Double		-
EnableVTControl	no	0, 1 0=VT off 1=VT on	-		-
Eject	no	-	-		-
Insert	no	-	-		-
SystemStatus	yes	-	-	Long Bit0=VT air status: 0=off, 1=on Bits1-2=Eject/Insert status: 0=inserted 1=ejecting 2=ejected 3=inserting Bits10-14=five air valves: 0=off, 1=on Bit10=VT air valve Bit11=bearing air valve Bit12=spin air valve Bit13=eject air valve Bit14=body air valve	-
ExitInterfaceApp	yes	-	-		-
ShowInterfaceWindow	yes	-	-		-
HideInterfaceWindow	yes	-	-		-
InterfaceWindowVisible	yes	-	true/ false		-

12.6

12.6 Automation Commands - Syntax & Examples
12.6.1 NTNMR Commands:

Abort

    Usage:

        objectname.Abort

    Object:

        Application, Document

    Description:

        Stops the current data acquisition immediately, without completing the current phase cycle.

    Example:

```
App.Abort
```

ActivateShims

    Usage:

        objectname.ActivateShims *shim_string*

    Object:

        Application

    Description:

        Activates the shims listed in *shim_string* for use in a subsequent auto-shim operation.

    Example:

```
App.ActivateShims "Z, X, ZX"
```

AddActiveDocToPrintPreview

    Usage:

        objectname.AddActiveDocToPrintPreview

    Object:

        Application

Description:

Adds the active data display to the print preview window. Use in conjunction with LoadPrintPreviewTemplate to automate creation of custom plots.

AddOneLineFit

Usage:

objectname.AddOneLineFit *start, stop, unit*

Object:

Application, Document

Description:

Adds a line fit simulation beginning at the left position specified in *start* and ending at the right position specified in *stop* with the units specified in *unit*. Valid units are "points", "hz", or "ppm". The line fit will not actually be visible unless the Line Fits Option is selected.

Example:

```
Data.AddOneLineFit  5.0, 4.5, "ppm"
```

AddParametersToPage

Usage:

objectname.AddParametersToPage *page_name*, *param_list*

Object:

Application

Description:

Adds the named parameters (*param_list*) to the named dashboard page (*page_name*). Parameters must be valid dashboard parameters and the list must be comma-delimited.

Example:

```
App.AddParametersToPage "Misc.", "Acq. Time"
```

AppendToFile

Usage:

objectname.AppendToFile *filename*, *overwrite*

Object:

Document, Application

Description:

Append data to a file specified by string sFileName; nOverWrite is an integer, when it's 0, just append the data, if it is 1, overwrite existing file

AutoBaselineFix

Usage:

objectname.AutoBaselineFix *"function", num_points, apply_to_all*

Object:

Application, Document

Description:

Performs an auto baseline fix on frequency domain data with the following parameters: *function* = "spline" or "polynomial"; *num_points* = indicates the number of points to include in the calculation; *apply_to_all* = true applies to all records in an nD set, false applies only to the active record.

AutoDisplayWhileProcessing

Usage:

objectname.AutoDisplayWhileProcessing=*true/false*

Object:

Document

Description:

Allows the data display to be disabled during automatic data processing.

AutoIntegrate
AI

Usage:

objectname. AutoIntegrate
objectname. AI

Object:

Application, Document

Description:

> Automatically integrates the spectrum based on peak finding variables set in the **Options|Integrate** mode. See the chapter on Options for more information.

AutoPeakPick

Usage:

> *objectname*.AutoPeakPick

Object:

> Application, Document

Description:

> Automatically peak picks the active spectrum and returns a string.

Example:

```
String_Peaks=App.AutoPeakPick
```

AutoPhase
AP

Usage:

> *objectname*.AutoPhase

Object:

> Application, Document

Description:

> Performs an auto phase of the type indicated in the 1D preferences (analytical or iterative).

BaselineCorrection
BC

Usage:

> *objectname*.BaselineCorrection
> *objectname*.BC

Object:

> Application, Document

Description:

Removes the dc level of the signal by calculating the mean over the last 12% of the data (Commonly applied before Fourier Transform is performed. Note that the NTNMR Fourier Transform *does not* apply any DC correction).

BaselineTilt
BT

Usage:

objectname.BaselineTilt
objectname.BT

Object:

Application, Document

Description:

Corrects for slope and offset of the baseline. (Commonly used before Auto Integrate.)

BlackmanHarris
BH

Usage:

objectname. BlackmanHarris *shift*, *width*
objectname.BH *shift*, *width*

Object:

Application, Document

Description:

Blackman Harris weighting using the phase shift and the function width. The *shift* and *width* parameters are optional. If they are omitted, the SB Shift and SB Width in the dashboard will be used.
Note: The *width* parameter specifies the **full** width of the function meaning that the function will cross zero at (Points 1D * 2).

CheckAcquisition

Usage:

objectname.CheckAcquisition

Object:

Application, Document

Description:

        Used to get the current acquisition status of NTNMR. The command is required in scripts that involve data acquisition. The CheckAcquisition command is used inside of a Do loop; the loop is not exited until CheckAcquisition returns a value of False.

Example:

```
Do While Not Data.CheckAcquisition
DoEvents
Loop
```

CheckShimProgress

Usage:

        objectname.CheckShimProgress

Object:

        Application

Description:

        Used to get the current acquisition status of an auto shim operation. The command should be used after starting an auto shim and before the next command. The CheckShimProgress command is used inside of a Do loop; the loop is not exited until CheckShimProgress returns a value of False.

Example:

```
Do While Not App.CheckShimProgress
DoEvents
Loop
```

ClearIntegrals

Usage:

        objectname.ClearIntegrals

Object:

        Application, Document

Description:

        Clears all integrals.

ClearPeakPick

Usage:

        objectname.ClearPeakPick

Object:
>   Application, Document

Description:
>   Clears all peak picks.

CloseActiveFile
Usage:
>   objectname.CloseActiveFile

Object:
>   Application

Description:
>   Close currently active NTNMR document

CloseFile
Usage:
>   objectname.CloseFile *sFilename*

Object:
>   Application

Description:
>   Close a specified NTNMR document by file name of type string

Compile
Usage:
>   objectname.Compile

Object:
>   Application, Document

Description:
>   Compile sequence, with return type true or false

CountShims
Usage:
>   objectname.CountShims

Object:
>   Application

Description:
>   Get the number of visible shims, return type is long integer

DigitalCorrect

Usage:

objectname.DigitalCorrect

Object:

Application, Document

Description:

Do an oneD conversion from Digital data to analog data

DoBaselineFix

Usage:

objectname.DoBaselineFix *function, num_points, "point1, point2, …", unit, apply_to_all*

Object:

Application, Document

Description:

Performs a baseline fix of type *function* ("spline" or "polynomial") with the number of points specified in *num_points*. The spectral positions of the points are specified in a comma delimited list as "*point1, point2, …*". The units for the list of points are specified as *unit* with valids units being "points", "hz", and "ppm". If *apply_to_all* is true, the baseline fix will be applied to all records. If the data is nD, if *apply_to_all* is false, the baseline fix will be applied to the active record only.

Example:

Data.DoBaselineFix "spline", 8, "100, 266, 511, 756, 1200, 1578, 2001, 2020", "points", false

DoubleExponentialMult
DM

Usage:

objectname. DoubleExponentialMult *value*
objectname. DM value

Object:

Application, Document

Description:

Performs a Gaussian/exponential weighting using *value*. The *value* parameter is optional; if omitted, DM 1D from Dashboard will be used.

EchoExponentialMult
EEM

Usage:

objectname. EchoExponentialMult *value*
objectname. EEM *value*

Object:

Application, Document

Description:

Performs an echo exponential multiplication on a 1D data set with a function defined by: $f(t) = e^{-t * value}$. The *value* parameter is optional; if omitted, LB 1D from Dashboard will be used. The time origin (t=0) is the center of the echo specified by the Dashboard parameter "Echo Center" (normally, but not necessarily, half the number of points).

EchoFT
EFT

Usage:

objectname. EchoFT
objectname. EFT

Object:

Application, Document

Description:

Performs a one dimensional fast Fourier Transform on a 1D data set, assuming that echo maximum, or time origin (t=0) is the center of the echo specified by the Dashboard parameter "Echo Center" (normally, but not necessarily, half the number of points).

EchoGaussianMultiplication
EGM

Usage:

objectname.EchoGaussianMultiplication
objectname.EGM

Object:

Application, Document

Description:

Performs a Gaussian multiplication on a 1D data set with a function defined by: $f(t) = \exp^{-t*value/1.6651}$. The *value* parameter is optional; if omitted, GM 1D from Dashboard is used. The time origin (t=0) is the center of the echo specified by the Dashboard parameter "Echo Center" (normally, but not necessarily, half the number of points).

EchoZeroFill
EZF

Usage:

objectname. EchoZeroFill
objectname. EZF

Object:

Application, Document

Description:

Doubles the number of data points (Points 1D) in memory and fills with zeros, taking into account that the data is in echo format by zero filling around the point specified by the Dashboard parameter "Echo Center." This command is valid only for 1D data sets.

ExistShim

Usage:

objectname.ExistShim *sShim*

Object:

Application

Description:

Test if the shim exists according to the shim name specified by sShim, which is a string, returns true or false

ExponentialMultiplication
EM

Usage:

objectname. ExponentialMultiplication *value*
objectname. EM *value*

Object:

Application, Document

Description:

Performs an exponential multiplication on the 1D data set with a function defined by: $f(t) = e^{-t * value}$. The *value* parameter is optional; if omitted, LB 1D from the Dashboard is used.

Export

Usage:

objectname.Export sFileName, nType

Object:

Application, Document

Description:

Save data as various type of file. SFileName will be the file name to be saved; nType = 0 saves text file, nType = 1 saves MacNMR file, nType = 2 saves bitmap file, nType = 3 saves jpeg file. Returns true or false

ExtractRecord

Usage:

objectname.ExtractRecord *n1DIndex, n2DIndex, n3DIndex, n4DIndex, sFileName*

Object:

Document, Application

Description:

Extract a particular record based on indexes and save record in the provided file. Indexes are long integers and sFileName is string. Return true or false

FitToWindow
FTW

Usage:

objectname.FitToWindow
objectname.FTW

Object:

Application, Document

Description:

Performs a one-time auto scale on the data display. If a zoom region is active, the data in the zoom will be auto scaled.

FourierTransform

FT

Usage:

objectname.FourierTransform

objectname.FT

Object:

Application, Document

Description:

Performs a one-dimensional fast Fourier transform on a 1D data set.

GaussianMultiplication

GM

Usage:

objectname.GaussianMultiplication

objectname.GM

Object:

Application, Document

Description:

Performs a Gaussian multiplication on a 1D data set with a function defined by: $f(t) = \exp^{-\ t*value\ /\ 1.6651}$. The *value* parameter is optional; if omitted, GM 1D from Dashboard is used.

Get1DRecord

Usage:

objectname.Get1DRecord *record_number*

Object:

Application, Document

Description:

Returns a variant that contains a 1D array of the data contained in the specified record in the format R, I, R, I, R, …

Example:

```
MyVariable = Data.Get1DRecord (5)
```

Get1DSelection

Usage:

objectname.Get1DSelection

Object:

    Application, Document

Description:

    Returns a string that contains the start and stop points of the currently highlighted 1D selection in the data display in the form (*start, stop*). If there is no currently highlighted region, Get1DSelection returns the first and last points in the spectrum.

Example:

```
MyVariable = Data.Get1DSelection
```

Get1DSelectionEnd

Usage:

    objectname.Get1DSelection

Object:

    Application, Document

Description:

    Returns the end point of the currently highlighted 1D selection in the data display.

Example:

```
MyVariable = Data.Get1DSelectionEnd
```

Get1DSize

Usage:

    objectname.Get1Dsize

Object:

    Document, Application

Description:

    Return 1D data size as a long integer

Get2DCursor

Usage:

    objectname.Get2DCursor *axis*

Object:

    Application, Document

Description:

> Returns a string that contains the point value for the current 2D cursor position for the specified axis. The axis is specified as an integer, 1or 2, where 1 is the horizontal axis and 2 is the vertical

Example:

```
MyVariable = Data.Get2Dcursor (2)
```

GetActiveDocPath
Usage:

> *objectname*.GetActiveDocPath

Object:

> Application, Document

Description:

> Application: Returns the full path the currently active data file. It is recommended that this command be used through the application whenever possible; since no previous link needs to be in place when accessing the application.

> Document: Returns the full path to the currently active data file. Note: Remember that a link to a data file must have been previously created before this command can be used. Once a link has been established to any Data file, this command can be used to recognize the front most Data file in NTNMR.

Application Example:

```
CreateObject ("NTNMR.Application")
Set App = GetObject(, "NTNMR.Application")
MyVariable = App.GetActiveDocPath
```

Document Example:

```
MyVariable = Data.GetActiveDocPath
```

GetActualShimPrecision
Usage:

> *objectname*.GetActualShimPrecision

Object:

Application

Description:

Returns the actual shim precision. Use this command to retrieve the actual shim precision to assess the quality of an auto shim operation.

Example:

```
MyVariable = Data. GetActualShimPrecision
```

GetComment

Usage:

objectname.GetComment

Object:

Application, Document

Description:

Returns the current contents of the data file comment field.

GetCursorPosition

Usage:

objectname.GetCursorPosition

Object:

Application, Document

Description:

Returns the point value of the current 1D cursor position as long integer.

Example:

```
MyVariable = Data.GetCursorPosition
```

GetData

Usage:

objectname.GetData

Object:

Application, Document

Description:

Returns a variant that contains a 1D array of the current data set in the format R, I, R, I, R, …

Example:

```
MyVariable = Data.GetData
```

GetDataPoint

Usage:

objectname.GetDataPoint *point_number*

Object:

Application, Document

Description:

Returns a variant that contains a 2-member array in the format R, I of the specified data point.

Example:

```
MyVariable = Data. GetDataPoint (64)
```

GetDataSize

Usage:

objectname.GetDataSize

Object:

Application, Document

Description:

Returns an integer that specifies the total number of points in the current data set (real + img.).

Example:

```
MyVariable = Data.GetDataSize
```

GetDocumentList

Usage:

objectname.GetDocumentList

Object:

Application

Description:

Returns a list of the currently open documents.

GetLatchedData

Usage:

objectname.GetLatchedData

Object:

Application

Description:

Get data from latched EM board and return a long integer.

GetNDDataPoint

Usage:

objectname.GetNDDataPoint *n1D, n2D, n3D, n4D*

Object:

Document

Description:

Returns a variant that contains a 2-member array in the format R,I of the data point specified by 1D,2D,3D and 4D indexes.

```
MyVariable = Data.GetNDDataPoint 32,2,1,1
```

GetNDRecordPointer

Usage:

objectname.GetNDRecordPointer *dimension*

Object:

Application, Document

Description:

Returns a long integer that specifies the active record for the dimension specified by the *dimension* parameter.

Example:

```
MyVariable = Data.GetNDRecordPointer (2)
```

GetNDSize

Usage:

objectname.GetNDSize *dimension*

Object:

Application, Document

Description:

Returns a long integer that specifies the number of real points in the specified dimension.

Example:

```
MyVariable = Data.GetNDSize (2)
```

GetNMRParameter

Usage:

objectname. GetNMRParameter *parameter_name*

Object:

Application, Document

Description:

Returns the variable associated with the Dashboard Item *parameter_name*. This command is commonly used to assign a Dashboard parameter value to a Visual Basic variable. The Dashboard parameter must be entered **exactly** as it appears in the Dashboard (case sensitive). Sequence variable values can also be retrieved.

Example:

```
MyVariable = Data.GetNMRParameter ("Points 1D")
```

GetOneShim

Usage:

objectname.GetOneShim sShim

Object:

Application

Description:

Get the value of the selected shim by shim name of type string, return type is long integer.

GetParameterListInPage

Usage:

objectname.GetParameterListInPage *page_name*

Object:

Application

Description:

Returns a comma de-limited list of the current parameters in the dashboard page specified by *page_name* in the format *"page_name = param1, param2, ..."*.

Example:

```
MyVariable = App.GetParameterListInPage ("Acquisition")
```

GetParameterPageList

Usage:

objectname.GetParameterPageList

Object:

Application

Description:

Returns a comma de-limited list of the current pages (i.e. "tabs") in the dashboard.

Example:

```
MyVariable = App.GetParameterPageList
```

GetPeakPickAmplitudes

Usage:

objectname.GetPeakPickAmplitudes

Object:

Application, Document

Description:

Returns a comma de-limited list of the current amplitudes for the current peak picks in the spectrum.

GetPeakPickList

Usage:

objectname.GetPeakPickList *units*

Object:

Application, Document

Description:

Returns a comma de-limited list of the current peak picks in the spectrum in units of *units* where *units* is "points", "hz", or "ppm".

GetReal

Usage:

objectname.GetReal

Object:

Application, Document

Description:

Get Real data from NTNMR data set. Need to create an Variant array with the size set as the number of real data, which is half of all data.

GetSequenceName

Usage:

objectname.GetSequenceName

Object:

Application, Document

Description:

Returns the full path to the sequence that is associated with the active data file.

Example:

```
MyVariable = Data.GetSequenceName
```

GetShims

Usage:

Objectname.GetShims

Object:

Application

Description:

Get an array filled with the name of all the visible shims, need to use a size-adjustable array to hold.

GetTable

Usage:

objectname.GetTable *table_name*

Object:

Application, Document

Description:

Returns the **name** and **contents** of the specified sequence table as a string in the format "*table_name = value1 value2 value3 ...*".

Example:

```
MyVariable = Data.GetTable
```

GetTableList

Usage:

objectname.GetTableList

Object:

Application, Document

Description:

Returns a space de-limited list of all table names in the sequence.

Example:

```
MyVariable = Data.GetTableList
```

Go

Usage:

objectname.Go

Object:

Application, Document

Description:

Starts a new acquisition using the sequence and parameters from the current window without zeroing the acquisition memory.

Example:

```
Data.Go
```

HideWindow

Usage:

objectname.HideWindow

Object:

Application

Description:

Minimizes AND hides NTNMR. NTNMR will not be visible in the Windows Task Bar after the Hide-Window command has been executed.

Example:

```
App.HideWindow
```

HilbertTransform

Usage:

objectname.HilbertTransform

Object:

Application, Document

Description:

Do Hilbert transform. Used to create a complex data set from a real data set (e.g. for phasing). Generally, the imaginary part should be zero before using this command.

Integrate

Usage:

objectname.Integrate *left_point, right_point, parameters*

Object:

Application, Document

Description:

Integrates the region of the spectrum as specified by *left_point* and *right_point*. The *parameters* string contains *units, offset, slope, curvature, multiplier* where: *units* = "points", "hz", or "ppm"; *offset* = the offset value; *slope* = the slope value; *curvature* = the curvature value; *multiplier* = the multiplier value. Consult the section of the manual on integration for information on the *offset, slope, curvature* and *multiplier* values.

Example:

Data.Integrate 1202, 1407, "points, 1, 1.2, 0.75, 2"

InverseFT
IFT

Usage:

objectname. InverseFT

Object:

Application, Document

Description:

Performs a one-dimensional inverse Fourier transform on a 1D data set.

Example:

Data.InverseFT

LeftShift

Usage:

objectname.LeftShift *points, true/false*

Object:

Application, Document

Description:

Left Shift the complex data by *points* with zeroing (*true*) or by circularly shifting the points to the end of the data (*false*). The *points* and *true/false* parameters are optional. If *points* is omitted, "Shift # pts" in the Dashboard will be used. If *true/false* is omitted, the shifted points will be zeroed. See the 1D Preferences for the option to circularly shift data points.

Example:

left shift 5 complex data points…

```
Data.LeftShift 5
```

Circularly left shift 5 complex data points…

```
Data.LeftShift 5, false
```

LoadIntegrals
Usage:

objectname.LoadIntegrals *file_path*

Object:

Application, Document

Description:

Loads the saved integral template specified in *filepath*.

Example:

```
Data.LoadIntegrals "C:\ntnmr\setup\int1.int"
```

LoadParameterSetupFromFile
Usage:

objectname.LoadParameterSetupFromFile *path_name*

Object:

Application

Description:

Loads the dashboard parameter setup file specified by *path name* and rearranges the dashboard accordingly. This command is a setup command *only*. It does not load parameter values. Use LoadParametersFromFile to load experiment parameters.

Example:

```
App.LoadParameterSetupFromFile "C:\ntnmr\setup\parameters.txt"
```

LoadParametersFromFile
Usage:

objectname.LoadParametersFromFile *path_name*

Object:

Application

Description:

Loads the experiment parameter setup file specified by *path name* and updates the dashboard. Only the parameters that are currently active in the dashboard will be displayed

Example:

App.LoadParametersFromFile "C:\ntnmr\setup\parameters.txt"

LoadPrintPreviewTemplate

Usage:

objectname.LoadPrintPreviewTemplate *path_name*

Object:

Application

Description:

Loads the print template file specified by *path_name*.

Example:

App. LoadPrintPreviewTemplate "C:\ntnmr\setup\stdplot.prt"

LoadSequence

Usage:

objectname.LoadSequence *path_name*

Object:

Application, Document

Description:

Loads the sequence specified by *path_name* and associates it and its variables with the active data file.

Example:

Data.LoadSequence ("C:\ntnmr\sequences\cosy.tps")

NDComplexTranspose

Usage:

objectname.NDComplexTranspose *nX, nY, nZ, nA*

Object:

Application, Document

Description:

Performs a complex transpose on the data, transposing the dimensions as specified by the *nX, nY, nZ,* and *nA* parameters. NDComplexTranspose moves the data in complex pairs.

Example:

Consider the data set:

| record 1 | R11 | I11 | R12 | I12 |
| record 2 | R21 | I21 | R22 | I22 |

that is stored by NTNMR as R11, I11, R12, I12, R21, I21, R22, I22, …

The command…

```
Data.NDComplexTranspose 2, 1, 3, 4
```

would result in the following matrix

| record 1 | R11 | I11 | R21 | I21 |
| record 2 | R12 | I12 | R22 | I22 |

NDFT

Usage:

objectname.NDFT

Object:

Application, Document

Description:

Executes the pre-defined set of processing commands currently set in the NDFT dialog. That can include baseline correction, zero filling, apodization, transformation and phasing. The command can be configured to process 1 to 3 dimensions with a single click. [SHIFT]-select the NDFT command or [shift]-click the NDFT toolbar button.

Example:

```
Data.NDFT
```

NDFTLoadTemplate
Usage:
> *objectname*.NDFTLoadTemplate *sFilePath*

Object:
> Application, Document

Description:
> Load a template NDFT file *.tft into NDFT dialog box.

Example:

```
Data.NDFTLoadTemplate  "C:\ntnmr\setup\template.tft"
```

NDTranspose
Usage:
> *objectname*.NDTranspose *nX, nY, nZ, nA*

Object:
> Application, Document

Description:
> Performs a hypercomplex transpose on the data, transposing the dimensions as specified by the *nX, nY, nZ,* and *nA* parameters.

Example:
> consider the data set:

record 1	R11	I11	R12	I12
record 2	R21	I21	R22	I22
record 3	R31	I31	R32	I32
record 4	R41	I41	R42	I42

> which is stored by NTNMR as R11, I11, R12, I12, R21, I21, R22, I22, …

> The command…

```
Data.NDTranspose 2, 1, 3, 4
```

> would result in the following matrix

record 1	R11	R21	R31	R41
record 2	I11	I21	I31	I41
record 3	R12	R22	R32	R42
record 4	I12	I22	I32	I42

OpenFile

Usage:

> *objectname*.OpenFile *file_path*

Object:

> Document, Application

Description:

> Opens the file specified in *file_path*.

Example:

Data.OpenFile "C:\ntnmr\data\c13.tnt"

OpenLockBar

Usage:

> objectname.OpenLockBar *lngOpen*

Object:

> Application

Description:

> Closes or opens the lock thermometer. 1/TRUE is to open,
> 0/FALSE is to close.

OpenTwoDView

Usage:

> *objectname*.OpenTwoDView *Init, Mode, Display, Aspect*

Object:

> Application, Document

Description:

> Opens the 2D display mode of NTNMR with the data display set
> according to the parameters *init*, *mode*, *display* and *aspect*.
> Init = True/False

True = use Mode, Disp, and Aspect information
False = use current NTNMR settings

Mode = 0, 1, 2, 3
    0 = Display real data
    1 = Display imaginary data
    2 = Display magnitude data
    3 = Display phase map

Disp = 0, 1, 2, 3
    0 = contour
    1 = gradient
    2 = stack plot
    3 = multi-image display

Aspect = 0, 1, 2
    0 = force square
    1 = by data
    2 = none

Example:

```
Data.OpenTwoDView 2, 1, 0
```

PhaseCorrection
PC

Usage:
> *objectname*.PhaseCorrection *ph0, ph1*
> *objectname*.PC *ph0, ph1*

Object:
> Application, Document

Description:
> Applies the specified zero (*ph0*) and first (*ph1*) order phase corrections to 1D data.

Example:

```
Data.PhaseCorrection 90, -180
```

PickOnePeak
Usage:
> *objectname*.PickOnePeak *position, unit*

Object:

Application, Document

Description:

Places a peak pick at the spectrum location specified in *position* with the units of the position value specified in *unit*. Valid units are "points", "hz" and "ppm".

Example:

```
Data.PickOnePeak 121.5, "ppm"
```

PrintFromPreview

Usage:

objectname.PrintFromPreview

Object:

Application

Description:

Sends the contents of the print preview window to the currently selected printer.

Example:

```
App.PrintFromPreview
```

Quit

Usage:

objectname.Quit

Object:

Application

Description:

Quits NTNMR. This command is the same as selecting Exit from the File menu.

Example:

```
App.Quit
```

ReadFirstHalf
Usage:
objectname.ReadFirstHalf

Object:
Application, Document

Description:
Extracts the first half of a data record. The Points 1D parameter on the dashboard is halved.

Example:

Data.ReadFirstHalf

ReadSecondHalf
Usage:
objectname.ReadSecondHalf

Object:
Application, Document

Description:
Extracts the second half of a data record. The Points 1D parameter on the dashboard is halved.

Example:

Data.ReadSecondHalf

RealFT
RFT
Usage:
objectname.RealFT
objectname.RFT

Object:
Application, Document

Description:
Calculates the one-sided Fourier transform where the imaginary part is set to zero. After the transform of n complex points, to get n/2 complex points back, use the ReadFirstHalf (or ReadSecondHalf) command. The transform is one-sided, 0 ->

+SW, with the first point at zero frequency, so the axis will have to be re-defined.

Example:

```
Data.RFT
Data.ReadFirstHalf
```

RemoveAllParameterPages
Usage:
> *objectname*.RemoveAllPages

Object:
> Application

Description:
> Removes all parameter pages (i.e. "tabs") from the dashboard. Note: NTNMR will automatically regenerate the Sequences tab as needed.

Example:

```
App.RemoveAllParameterPages
```

RemoveParameterPage
Usage:
> *objectname*.RemoveParameterPage *page_name*

Object:
> Application

Description:
> Removes the parameter page specified parameter page from the dashboard.

Example:

```
App.RemoveParameterPage "Acquisition"
```

RepeatScan
Usage:
> objectname.RepeatScan

Object:
> Application

Description:

Save the current scan counters and do ZG

Reset

Usage:

objectname.Reset

Object:

Application, Document

Description:

Executes the reset hardware command.

ResetDimensionInfo

Usage:

objectname.ResetDimensionInfo *num_points_X, num_points_Y, num_points_Z, num_points_A*

Object:

Application, Document

Description:

Re-defines the dimension information for a data set by specifying the number of points for each dimension. The number of points for the first ("X") dimension is specified as complex, all other dimensions are specified as total points. The total number of points in a data set CANNOT be changed using this command, only the dimension breaks are changed. An error will result if the total number of points specified by the ResetDimensionInfo command does not match the total number of points in the data set.

Example:

Consider a data set that contains 32 records with each record having 512 points.

```
Data.ResetDimensionInfo 128, 128, 1, 1
```

The resulting data would contain 128 records with each record having 128 points.

ReverseSpectrumColumns

Usage:

objectname.ReverseSpectrumColumns

Object:

Application, Document

Description:

Performs a reverse spectrum on the indirect dimension (displayed vertically in NTNMR) of a 2D data set.

Example:

```
Data.ReverseSpectrumColumns
```

ReverseSpectrumRows

Usage:

objectname.ReverseSpectrumRows

Object:

Application, Document

Description:

Performs a reverse spectrum on the direct dimension (displayed horizontally in NTNMR) of a 2D data set.

Example:

```
Data.ReverseSpectrumRows
```

RightShift

Usage:

objectname.RightShift *points, true/false*

Object:

Application, Document

Description:

Right Shift the complex data by *points* number of points with zeroing (*true*) or by circularly shifting the points to the end of the data (*false*). The *points* and *true/false* parameters are optional. If *points* is omitted, ìShift # ptsî in the Dashboard will be used. If *true/false* is omitted, the shifted points will be zeroed. See the 1D Preferences for the option to circularly shift data points.

Example:

Right shift 5 complex data points

```
Data.RightShift 5, true
```

Circularly right shift 5 complex data points

```
Data.RightShift 5, false
```

Save

Usage:

objectname.Save

Object:

Application, Document

Description:

Saves the current data under the same filename replacing the original data file.

Example:

```
Data.Save
```

SaveAllIntegrals

Usage:

objectname.SaveAllIntegrals *sFileName*

Object:

Application, Document

Description:

Save All integrals to the file specified by sFileName, the file should be .int file that can be loaded by NTNMR.

SaveAs

Usage:

objectname.SaveAs *file_path*

Object:

Application, Document

Description:

Saves the current data under the filename specified in *file_path*.

Example:

```
Data.SaveAs "C:\mydatafolder\sample1\72397.tnt"
```

SaveAsJPG

Usage:

objectname.SaveAsJPG *sFileName, nWidth, nHeight*

Object:

Application, Document

Description:

Save data as a nWidth*nHeight JPEG file specified by string

SaveIntegrals

Usage:

objectname.SaveIntegrals *file_path*

Object:

Application, Document

Description:

Saves the current integrals under the filename specified in *file_path*.

Example:

```
Data.SaveIntegrals "C:\ntnmr\setup\int1.int"
```

SaveIntegralsAsText

Usage:

objectname.SaveIntegralsAsText *sFileName*

Object:

Application, Document

Description:

Save integrals as a text file specified by sFileName

SaveParameterSetupToFile

Usage:

objectname.SaveParameterSetupToFile *file_path*

Object:

Application

Description:

Saves the current dashboard setup to the filename specified in *file_path*. This command DOES NOT SAVE parameter values, only the current dashboard setup.

Example:

```
App.SaveParameterSetupToFile "C:\ntnmr\setup\mysetup.txt"
```

SaveParametersToFile

Usage:

objectname.SaveParametersToFile *file_path*

Object:

Application

Description:

Saves the current parameters from the dashboard to the specified file.

Example:

```
App.SaveParametersToFile "C:\ntnmr\setup\c13.txt"
```

SavePeakPickList

Usage:

objectname.SavePeakPickList *file_path*

Object:

Application, Document

Description:

Saves the current peak pick information in a text file as indicated by *file_path.*

Example:

```
Data.SavePeakPickList "C:\mydata\sample1\sample1_peaks.txt"
```

SaveSequence

Usage:

objectname.SaveSequence

Object:

Application, Document

Description:

Saves the current Sequence under the same filename replacing the original sequence file.

Example:

```
Data.SaveSequence
```

SaveSequenceAs

Usage:

objectname.SaveSequenceAs *file_path*

Object:

Application, Document

Description:

Saves the current Sequence under the filename specified in *file_path*.

Example:

```
Data.SaveSequenceAs "C:\mydatafolder\sample1\cosy.tps"
```

SaveShims

Usage:

objectname.SaveShims

Object:

Application

Description:

Saves the current shim values into the default.shm.

Example:

```
App.SaveShims
```

Set2DCursor

Usage:

objectname.Set2DCursor *axis, point_number*

Object:

Application, Document

Description:

Sets the 2D cursor for the axis specifed by *axis* to the point value specified by *point_number*. 1 = hoizontal axis, 2 = vertical axis.

Example:

```
Data.Set2DCursor 2, 112
```

Set2DNumPanes

Usage:

objectname.Set2DNumPanes *nPanes*

Object:

Document, Application

Description:

Set the number of panes for 2D data by integer nPanes, return true or false

SetActiveParameterPage

Usage:

objectname.SetActiveParameterPage *page_name*

Object:

Application

Description:

Forces NTNMR to display the Dashboard page specified by the parameter *page_name*.

Example:

```
App.SetActiveParameterPage "Acquisition"
```

SetAutoShimParameters

Usage:

objectname.SetAutoShimParameters *sDelay, dbPrecision, nType, nStep*

Object:

Application

Description:

Set AutoShimParameters according to sDelay, which is a string representing delay in seconds; dbPrecision is a double number for target precision; nType is 0 for LOCK and 1 for FID; nStep is for shim step

SetComment

Usage:

objectname.SetCursorPoitions *comment*

Object:

Application, Document

Description:

Sets the data file comment to the text specified in *comment*.

Examples:

```
Data.SetComment "C13, July 7, 1998"
```

```
Dim myComment as String
myComment = "C13 Exp." & vbcrlf & "July 7, 1998"
Data.SetComment myComment
```

SetCursorPosition

Usage:

objectname.SetCursorPoitions *point_value*

Object:

Application, Document

Description:

Sets the 1D cursor to the data point specified by the parameter *point_value*.

Example:

```
Data.SetCursorPosition 2012
```

SetData

Usage:

objectname.SetData *numPoints*, *array*

Object:

Application, Document

Description:

Sets the data in the active object (data file) to the values contained in the parameter *array*, which is a 1D array of the format R, I, R, I, R… Note that if the data specified in the parameter *array* is larger than the current data buffer, NTNMR will return an error. If the data size specified in the parameter *array* is smaller than the current data buffer, only the portion by *array* will be replaced. The parameter *numPoints* is the total number of points, *not* the number of complex points.

Example:

```
Data.SetData 512, myArray
```

SetDataPoint
Usage:

objectname.SetDataPoint *point_number, array*

Object:

Application, Document

Description:

Sets the data at the point specified by *point_number* in the active object (data file) to the values contained in the parameter *array*, which is a 1D array of the format R, I.

Example:

```
Data.SetDataPoint 5, my2PointArray
```

SetLatchedData
Usage:

Objectname.SetLatchedData *nData*

Object:

Application

Description:

Set data to latched EM board with an input of long type

SetNDApodization
Usage:

objectname.SetNDApodization *dimension, par1_value, par2_value[optional] , par3_value [optional] , par4_value [optional]*

Object:
>   Application, Document

Description:
>   Sets the apodization values to be used with the apodization
>   function specified by the SetNDParameters command for the
>   specified dimension. This is an initialization command that is
>   used in conjunction with SetNDParameters and NDFT. The
>   values specified by SetNDApodization are applied *only* if
>   SetNDParameters is called with the *apodization* parameter set to
>   1 through 9 followed by the NDFT command. Parameters
>   indicated as [optional] are required only if the apodization
>   function to be called require them. See SetNDParameters for
>   more information and examples.
>
>   Dimension = 1, 2, 3, 4
>   >   1 = first dimension
>   >   2 = second dimension
>   >   3 = third dimension
>   >   4 = fourth dimension
>
>   par1_value = *value*
>   >   *value* = apodization parameter
>
>   par2_value = *value*
>   >   *value* = apodization parameter
>
>   par3_value = *value*
>   >   *value* = apodization parameter
>
>   par4_value = *value*
>   >   *value* = apodization parameter

SetNDEchoCenter

Usage:
>   *objectname*.SetNDEchoCenter *Dimension, center_value*

Object:
>   Application, Document

Description:
>   Sets the echo center to be used by the EchoFT and Echo-mode
>   apodization functions for the specified dimension. This is an

initialization command that is used in conjunction with SetNDParameters and NDFT. The value specified by SetNDEchoCenter are applied *only* if SetNDParameters is called with the *FT* parameter set to 4, indicating an Echo FT, followed by the NDFT command. See SetNDParameters for more information and examples.

Dimension = 1, 2, 3, 4
      1 = first dimension
      2 = second dimension
      3 = third dimension
      4 = fourth dimension

cnter_value = *value*
      value = point number of the echo center

SetNDParameters

Usage:
      objectname.SetNDParameters *Dimension, BC, ZeroFill, FT, Reverse, Phase, apodization*

Object:
      Application, Document

Description:
      Initializes the parameters to be used with the NDFT command. This command is used in conjunction with the SetNDPhase, SetNDEchoCenter, SetNDApodization and NDFT commands. The following parameters are used:

Dimension = 1, 2, 3, 4
      1 = first dimension
      2 = second dimension
      3 = third dimension
      4 = fourth dimension

BC = true/false
      true = apply baseline correction
      false = no baseline correction

ZeroFill = 0, 1, 2…
      value = number of zero fills

FT = 0, 1, 2, 3, 4

0 = none
1 = FFT
2 = Inverse FFT
3 = Real FFT
4 = Echo FFT

Reverse = true/false
true = apply spectrum reverse
false = no spectrum reverse

Phase = 0, 1, 2
0 = none
1 = use values specified by SetNDPhase or set in the NDFT dialog in NTNMR.
2 = use "system" phase parameters

Apodization = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
0 = none
1 = EM
2 = Gaussian
3 = double exponential
4 = sine bell
5 = sine bell squared
6 = Blackman Harris
7 = trapezoid
8 = traf
9 = echo exponential multiplication

Example:
The following example initializes the NDFT template to apply a baseline correction and courier transform to a two dimensional data set using a standard complex transpose.

```
Data.SetNDTranspose 2, 0, 1
Data.SetNDParameters 1, true, 0, 1, false, 0, 0
Data.SetNDParameters 2, true, 0, 1, false, 0, 0
Data.NDFT
```

SetNDPhase
Usage:
objectname.SetNDPhase *Dimension, ph0_value, ph1_value*

Object:
Application, Document

Description:

Sets the zero and first order phase constants to be applied for the specified dimension. This is an initialization command that is used in conjunction with SetNDParameters and NDFT. The values specified by SetNDPhase are applied *only* if SetNDParameters is called with the *phase* parameter set to 1 followed by the NDFT command. See SetNDParameters for more information and examples.

Dimension = 1, 2, 3, 4
    1 = first dimension
    2 = second dimension
    3 = third dimension
    4 = fourth dimension

ph0_value = *value*
    value = zero order phase correction

ph1_value = *value*
    value = first order phase correction

SetNDRecordPointer

Usage:

objectname.SetNDRecordPointer *dimension, record_number*

Object:

Application, Document

Description:

Sets the NTNMR data display to data specified by the *dimension* and *record_number* parameters.

Example:

```
Data.SetNDRecordPointer 3, 12
```

SetNDTranspose

Usage:

objectname.SetNDTranspose *Dimension, Type, Data*

Object:

Application, Document

Description:

Sets the transpose type and final data for the specified dimension. This is an initialization command that is used in conjunction with NDFT. The specified transpose is applied *only* by the NDFT command. See SetNDParameters for more information and examples.

Dimension = 1, 2, 3, 4

1 = first dimension
2 = second dimension
3 = third dimension
4 = fourth dimension

Type = 0, 1

0 = complex
1 = hypercomplex

Data = 0

0 = keep all data
1= keep real

SetNMRParameter

Usage:

objectname.SetNMRParameter *"parameter_name", value*

Object:

Application, Document

Description:

Sets the Dashboard Item specified in *parameter_name* to the value set in *value*.

Example:

Data.SetNMRParameter "Points 1D", 1024

SetOneShim

Usage:

objectname.SetOneShim *"shim_name", value*

Object:

Application

Description:

Sets the specified in *shim_name* to the value set in *value*.

Example:

> App.SetOneShim "x", 1000

SetParameterListInPage

Usage:

objectname.SetParameterListInPage *"page_name", "par1, par2, par3"*

Object:

Application

Description:

Sets the list of parameters to be displayed in the dashboard page specified by *page_name*. The dashboard page must exist and the comma de-limited list of dashboard items must be valid. Parameters can, however, be from any page.

Example:

> App.SetParameterListInPage   "Acquisition",   "Nucleus,   Obs. Freq., LB1D"

SetRecordPointer

Usage:

objectname.SetRecordPointer *2D_record_number*

Object:

Application, Document

Description:

Sets the NTNMR display to the 2D record number specified by the parameter *2D_record_number*.

Example:

> Data.SetRecordPointer 30

SetSelection

Usage:

objectname.SetSelection *point_start, point_end*

Object:

Application, Document

Description:

Highlights the region of data in the 1D display area as specified by the parameters *point_start* and *point_end*. This command is equivalent to a click-drag-release operation in the 1D display area.

Example:

```
Data.SetSelection 100, 200
```

SetTable

Usage:

objectname.SetTable *"table_name", "value1 value2 value3 ..."*

Object:

Application, Document

Description:

Sets the contents of the table specified by *table_name* to the values specified by "*value1 value2 value3 ...*". Note: tables entries should be [space] de-limited.

Example:

```
Data.SetTable "ph0", "1 2 3 0"
```

ShowIntegrals

Usage:

objectname.ShowIntegrals *true/false*

Object:

Application, Document

Description:

Shows (true) and hides (false) current integrals.

Example:

```
App.ShowIntegrals false
```

ShowLineFits
> Usage:
>> objectname.ShowLineFits bShow
>
> Object:
>> Application, Document
>
> Description:
>> toggle display of line fits, according to Boolean variable bShow

ShowPeakPicks
> Usage:
>> *objectname*.ShowPeakPick *true/false*
>
> Object:
>> Application, Document
>
> Description:
>> Shows (true) and hides (false) the current peak picks.
>
> Example:

```
App.ShowPeakPick true
```

ShowWindow
> Usage:
>> *objectname*.ShowWindow
>
> Object:
>> Application
>
> Description:
>> Activates the main NTNMR application window after it has been minimized by clicking on the minimize button in the applications title bar or by executing the Hide Window command.
>
> Example:

```
App.ShowWindow
```

SinBell
SB
> Usage:
>> *objectname*.SinBell (*shift, width, skew*)
>> *objectname*.SB (*shift, width, skew*)

Object:

> Application, Document

Description:

> Performs a Sine Bell apodization on 1D data using the parameters *shift, width* and *skew*. The *shift, width* and *skew* parameters are optional. If they are omitted, SB Shift, SB Width and SB Skew in the dashboard will be used. Note: The *width* parameter specifies the full width of the function meaning that the function will cross zero at (Points 1D * 2).

Example:

> | Data.SinBell 90, 8192, 1 |
> |---|

SinBellSquared
SBS

Usage:

> *objectname*.SinBellSquared *shift, width, skew*
> *objectname*.SBS *shift, width, skew*

Object:

> Application, Document

Description:

> Performs a Sine Bell Squared apodization on 1D data using the parameters *shift, width,* and *skew*. The *shift, width* and *skew* parameters are optional. If they are omitted, SB Shift, SB Width and SB Skew in the dashboard will be used. Note: The *width* parameter specifies the half width of the function meaning that the function will cross zero at (Points 1D).

Example:

> | Data.SinBellSquared 90, 8192, 1 |
> |---|

SortDataPoints

Usage:

> objectname.SortDataPoints nPattern

Object:

> Application, Document

Description:

Unshuffle interleaved data points by integer *nPattern*. Especially useful for processing data from multiple receivers.

```
Data.SortDataPoints 4
```

StartShims

Usage:

objectname.StartShims

Object:

Application

Description:

Initiates an auto shim. Use in conjunction with ActivateShims and CheckShimProgress.

Example:

```
App.ActivateShims "x, y, z"

App.StartShims

Do While Not App.CheckShimProgess
DoEvents
Loop
```

Stop

Usage:

objectname.Stop

Object:

Application, Document

Description:

Stops the current data acquisition after completion of the current phase cycle.

Example:

```
App.Stop
```

TextToTime

Usage:

Objectname.TextToTime sTime

Object:

> Application, Document

Description:

> Convert text string "sTime" to time in seconds. The input string may contain time, ends in units "n,u,m,s" 's' = default

TrafWindow
tw

Usage:

> *objectname*.TrafWindow *value*
> *objectname*.TW *value*

Object:

> Application, Document

Description:

> Performs Traficante Ziessow weighting (J. Mag. Res., 66, 182-186 (1986)) using the parameter *value*. The *value* parameter is optional. If omitted, the Dashboard parameter Traf 1D will be used.

Example:

> | Data.TrafWindow 5 |
> | --- |

Transpose

Usage:

> *objectname*.Transpose

Object:

> Application, Document

Description:

> Performs a complex transpose on the X and Y dimensions of a 2D data set.

Example:

> | Data.Transpose |
> | --- |

Trapezoid
TZ

Usage:

> *objectname*.Trapezoid *value1, value2, value3, value4*

objectname.TZ *value1, value2, value3, value4*

Object:

Application, Document

Description:

Performs weighting using the 4 data points *value1, value2, value3 and value 4* as the trapezoid shape de-limiters. The parameters *value1 - 4* are optional. If omitted, the Dashboard parameters TZ 1 1D, TZ 2 1D, TZ 3 1D and TZ 4 1D will be used.

Example:

```
Data.TZ 2, 5, 512, 1024
```

UpdateDisplay

Usage:

objectname.UpdateDisplay

Object:

Application, Document

Description:

Forces a redraw of the 1D NTNMR data display area. This command can be used when AutoDisplayWhileProcessing has been set to false to force redraws at specific times.

Example:

```
Data.UpdateDisplay
```

ZeroFill
ZF

Usage:

objectname.ZeroFill *value*
objectname.ZF *value*

Object:

Application, Document

Description:

Zero fills the data to the number of points specified by *value*. The *value* parameter is optional. If omitted, the data size will be doubled.

Example:

```
  Data.ZeroFill 8192
```

ZeroAndGo
ZG
Usage:
objectname.ZeroAndGo
objectname.ZG

Object:
Application, Document

Description:
Zeros the signal averager memory and starts an acquisition. Use in conjunction with the CheckAcquisition command.

Example:

```
  Data.ZG
   Do While Not Data.CheckAcquisition
  Loop
```

ZeroImaginary
Usage:
objectname.ZeroImaginary nRecord

Object:
Application, Document

Description:
According to nRecord which is a long integer, make imaginary data of that record zero

Zoom
Usage:
objectname.Zoom *point_start, point_end*

Object:
Application, Document

Description:
Executes a zoom command on the 1D data display using the parameters *point_start* and *point_end*.

Example:

```
Data.Zoom 100, 790
```

12.6.2 SampleInterface Commands:

ExitInterfaceApp
Usage:
Objectname.ExitInterfaceApp

Object:
Application

Description:
Used to exit and unload the application. This Method needs no parameter and returns nothing.

HideInterfaceWindow
Usage:
Objectname.HideInterfaceWindow
Object:
Application

Description:
Used to hide the application window without unloading it. This Method needs no parameter and returns nothing.

Insert
Usage:
Objectname.Insert

Object:
Application

Description:
Used to send insert sample command. This Method needs no parameter and returns nothing.

InterfaceWindowVisible
Usage:
blnWinVisible=*Objectname*.InterfaceWindowVisible

Object:
Application

Description:

Used to check the application window status. This Read property needs no parameters and returns a Boolean value True or False. True means the application window is visible or vice versa.

SampleTemperature

Usage:

dblVTTemp=*Objectname.*SampleTemperature

Object:

Application

Description:

Used to get VT temperature. This Read property returns a double value and needs no parameter.

ShowInterfaceWindow

Usage:

*Objectname.*ShowInterfaceWindow

Object:

Application

Description:

Used to show and load the application window. This Method needs no parameter and returns nothing.

SpinRate

Usage:

dblSpinRate=*Objectname.*SpinRate

Object:

Application

Description:

Used to get the current spin rate. This Read property returns a double floating point value and needs no parameter.

Example:

```
Dim dblSpinRate as double
dblSpinRate=App.SpinRate
```

SpinRateSetPoint

Usage:

Objectname.SpinRateSetPoint =dblSpinSetPoint

Object:

Application

Description:

Used to send spin set point. This Write property needs a double value parameter and returns nothing.

Example:

```
App.SpinRateSetPoint =20.0
```

SpinSample

Usage:

Objectname.SpinSample intSpinMode

Object:

Application

Description:

Used to set spin mode. This Method needs an integer parameter 0, 1, or 2 and returns nothing. intSpinMode=0 to turn off spin; 1 to turn on spin with closed loop; 2 to turn on spin with open loop.

Example:

```
To turn on spin with closed loop:
App.SpinSample 1
```

SystemStatus

Usage:

lngStatus=Objectname. SystemStatus

Object:

Application

Description:

Used to check the system status bits. This Read property needs no parameter and returns a long integer.
Bit0=VT air status: 0=off, 1=on
Bits1-2=Eject/Insert status: 0=inserted

<div align="center">
1=ejecting
2=ejected
3=inserting
</div>

Bits10-14=five air valves: 0=off, 1=on
Bit10=VT air valve
Bit11=bearing air valve
Bit12=spin air valve
Bit13=eject air valve
Bit14=body air valve

VTSetPoint

Usage:

Objectname.VTSetPoint =dblVTSetPoint

Object:

Application

Description:

Used to send VT set point. This Write property needs a double value parameter and returns nothing.

12.6.3 TecmagLock Commands:

GetLockState

Usage:

intLockState=*Objectname*.GetLockState

Object:

Application

Description:

Used to get Lock State. This Read property needs no parameter and returns an integer value.

ExitLockApp

Usage:

Objectname.ExitLockApp

Object:

Application

Description:

Used to exit and unload the application. This Method needs no parameter and returns no parameters.

LockSweepSpeed

Usage:

lngSwpSpd=*Objectname*.LockSweepSpeed

Object:

Application

Description:

Used to get Lock Sweep Speed. This Read property needs no parameter and returns a long integer value.

LockSweepSpeed

Usage:

Objectname.LockSweepSpeed= lngSwpSpd

Object:

Application

Description:

Used to set Lock Sweep Speed. This Write property needs a long integer value parameter and returns nothing.

Example:

```
App.LockSweepSpeed=200
```

LockSweepWidth

Usage:

lngSwpWidth=*Objectname*.LockSweepWidth

Object:

Application

Description:

Used to get Lock Sweep Width. This Read property needs no parameter and returns a long integer value.

LockSweepWidth

Usage:

Objectname.LockSweepWidth= lngSwpWidth

Object:

Application

Description:

Used to set Lock Sweep Width. This Write property needs a long integer value parameter and returns nothing.

Example:

```
App.LockSweepWidth=100
```

LockFieldCenter

Usage:

sngFldCntr=*Objectname*.LockFieldCenter

Object:

Application

Description:

Used to get Lock Field Center. This Read property needs no parameter and returns a single floating-point value.

LockFieldCenter

Usage:

Objectname.LockFieldCenter= sngFldCntr

Object:

Application

Description:

Used to set Lock Field Center. This Write property needs a single floating-point parameter and returns nothing.

Example:

```
App.LockFieldCenter=10.0
```

LockSolventPPM

Usage:

sngSolPPM=*Objectname*.LockSolventPPM

Object:

Application

Description:

Used to get Lock Solvent PPM. This Read property needs no parameter and returns a single floating-point value.

LockSolventPPM

Usage:

    Objectname.LockSolventPPM= sngSolPPM

Object:

    Application

Description:

    Used to set Lock Solvent PPM. This Write property needs a single floating-point parameter and returns nothing.

LockSolventText

Usage:

    strSolTxt=*Objectname*.LockSolventText

Object:

    Application

Description:

    Used to get Lock Solvent Text. This Read property needs no parameter and returns a string.

LockSolventText

Usage:

    Objectname.LockSolventText =strSolTxt

Object:

    Application

Description:

    Used to set Lock Solvent Text. This Write property needs a string parameter and returns nothing.

LockTxPower

Usage:

    sngTxPwr=*Objectname*.LockTxPower

Object:

    Application

Description:

    Used to get Lock Tx Power. This Read property needs no parameter and returns a single floating-point value.

LockTxPower

Usage:

Objectname. LockTxPower= sngTxPwr

Object:

Application

Description:

Used to set Lock Tx Power. This Write property needs a single floating-point parameter and returns nothing.

LockRxGain

Usage:

sngRxGain=*Objectname*.LockRxGain

Object:

Application

Description:

Used to get Lock Rx Gain. This Read property needs no parameter and returns a single floating-point value.

LockRxGain

Usage:

Objectname.LockRxGain =sngRxGain

Object:

Application

Description:

Used to set Lock Rx Gain. This Write property needs a single floating-point parameter and returns nothing.

LockNMRPhase

Usage:

sngFldCntr=*Objectname*.LockNMRPhase

Object:

Application

Description:

Used to get Lock NMR Phase. This Read property needs no parameter and returns a single floating-point value.

LockNMRPhase

Usage:

*Objectname.*LockNMRPhase= sngFldCntr

Object:

Application

Description:

Used to set Lock NMR Phase. This Write property needs a single floating-point parameter and returns nothing.

LockLevel

Usage:

sngLockLvl=*Objectname.*LockLevel

Object:

Application

Description:

Used to get Lock Level. This Read property needs no parameter and returns a single floating-point value.

LockLoopChar

Usage:

*Objectname.*LockLoopChar= intLoopChar

Object:

Application

Description:

Used to set Lock Loop Character. This Write property needs an integer value parameter and returns nothing.

ShowLockWindow

Usage:

*Objectname.*ShowLockWindow

Object:

Application

Description:

Used to show and load the application window. This Method needs no parameter and returns no parameters.

StartLock

Usage:

Objectname.StartLock

Object:

Application

Description:

Used to start Lock process command. This Method needs no parameter and returns nothing.

StartAutoPhase

Usage:

Objectname.StartAutoPhase

Object:

Application

Description:

Used to start Auto Phase process command. This Method needs no parameter and returns nothing.

StartAutoLock

Usage:

Objectname.StartAutoLock

Object:

Application

Description:

Used to start Auto Lock process command. This Method needs no parameter and returns nothing.

StartHold

Usage:

Objectname.StartHold

Object:

Application

Description:

Used to start Hold process command. This Method needs no parameter and returns nothing.

StartSweep
Usage:
>> *Objectname*.StartSweep

Object:
>> Application

Description:
>> Used to start Sweep process command. This Method needs no parameter and returns nothing.

StopLock
Usage:
>> *Objectname*.StopLock

Object:
>> Application

Description:
>> Used to stop Lock process command. This Method needs no parameter and returns nothing.

12.7 Writing NMRScripts in VBScript

12.7

The VBScript language is a subset of Microsoft's Visual Basic programming language. Through the use of the "Windows Scripting Host" or "WSH," VBScript can be used as a system level scripting language. Since VBScript has very limited user interface tools, VBScripts written to control NTNMR are more like macros than Visual Basic programs.

While VBScripts can be written using any text editor, the NMRScriptEditor provided with NTNMR is an excellent tool. The NMRScriptEditor provides several advantages over a standard text editor:

- Built in NTNMR language reference with syntax examples
- Pre-written NTNMR communication header
- Execute a script from the editor without the need to first save to disk

The NMRScriptEditor (shown below) can be launched by selecting it from the Scripts menu in NTNMR or by navigating to the script sub-directory of the NTNMR directory and double-clicking.
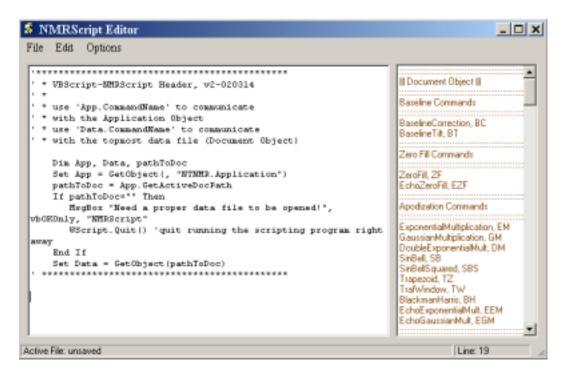
Figure 12.2 NMRScript Editor window

When the NMRScriptEditor is launched, a new script file is automatically created with the NTNMR "communication" header inserted. The

communication header is required to set up an OLE link with NTNMR. The default header establishes communication with the top most document in NTNMR as the *Document* object, which is assigned to the variable called *Data*. Using the default header will allow any script that is written to automatically act on the top most data file open in NTNMR.

Once the NMRScriptEditor is opened, commands can be added by typing below the header or by double-clicking a command in the right hand NTNMR command reference pane. Note also that selecting and pausing over a command will pop up a tool tip with a syntax example for that command.

Users may wish to install the VBScript language reference from the NTNMR CD or connect to http://www.microsoft.com/scripting for the VBScript language reference (choose *documentation*, then *script technologies*, and finally *VBScript*. Additional information sources are mentioned at the end of this chapter in the References section.
An especially useful feature is the File System Object (http://msdn.microsoft.com/library/en-us/script56/html/sgFileSystemObjectModel.asp). This allows you to manipulate files and directories, and to create or read simple text files, for example to store a list of results for later analysis.

12.8 VBScript Examples

12.8

12.8.1 Example 1: Auto Process 1D

```
'communication header
' *****************************************
    Dim App, Data, pathToDoc
    Set App = GetObject(, "NTNMR.Application")
    pathToDoc = App.GetActiveDocPath
    Set Data = GetObject(pathToDoc)
' *****************************************

'script
 Data.BC
 Data.EM 1
 Data.FT
 Data.AP
Data.FitToWindow
```

12.8.2 Example 2: Chirp Generator

```
'communication header
' ******************************************
    Dim App, Data, pathToDoc
    Set App = GetObject(, "NTNMR.Application")
    pathToDoc = App.GetActiveDocPath
    Set Data = GetObject(pathToDoc)
' ******************************************

'script
dim phase(2048)
titlestring = "NTNMR Chirp Generator"
DeltaT = (inputbox( "Time per point (microsec)", titlestring, "5"))
DeltaT = DeltaT * 1.0e-6
NPts = (inputbox( "Number of points (2-2047)",titlestring,"1024"))
SW = (inputbox( "Sweep Width (Hz)",titlestring,"5000"))

' Check for valid entries
if NPts > 2047 then
  msgbox "Too Many Points !!!",vbCritical
  wscript.quit
end if

' OK... proceed

phase(i)=0
for i = 1 to NPts
  freq = (i-NPts/2)*SW/NPts
  inc = 2 * 3.14159 * freq * DeltaT
  phase(i) = (phase(i-1) + inc) '
  'wscript.echo  phase(i)
  phasetemp = (phase(i) * 360./(2 * 3.14159)) mod 360
  'tempstring = tempstring & " " & (phase(i)) & chr(13) & chr(10)
  tempstring = tempstring & " " & (phasetemp) & chr(13) & chr(10)
next

'msgbox tempstring

tablename = inputbox( "Sequence Table Name",titlestring,"phase0")

Data.SetTable tablename, tempstring
```

12.9

12.9 Visual Basic Terminology

While it is beyond the scope of this manual to describe the details of programming in Visual Basic, the following discussion of terminology will facilitate the material presented here.

Control

A control is a tool you use to create objects on a Visual Basic form. You select controls from the toolbox and use the mouse to draw objects on a form. You use most controls to create user interface elements, such as command buttons, image boxes and list boxes.

Object

An object is a type of user interface element you create on a Visual Basic form by using a toolbox control. (In fact, in Visual Basic, the form itself is also an object.) You can move, resize and customize objects by using property settings. Objects also have what is known as inherent functionality-they know how to operate and can respond to certain situations on their own. (A list box "knows" how to scroll, for example.) You can customize Visual Basic objects by using event procedures that are fine-tuned for different conditions in a program.

Property

A property is a value or characteristic held by a Visual Basic object, such as Caption or ForeColor. Properties can be set at design time by using the Properties window or at <u>runtime</u> by using statements in the program code. In code, the format for setting a property is:

- Object.Property = Value

    Where
- *Object* is the name of the object you're customizing.

- *Property* is the characteristic you want to change.

- *Value* is the new property setting.

For example,
- Command1.Caption = "Hello"

Could be used in the program code to set the Caption property of the Command1 object to "Hello."

Event Procedure

An event procedure is a block of code that runs when a program object is manipulated. For example, clicking the first command button in a program executes the Command1_Click event procedure. Event procedures typically evaluate and set properties and use other program statements to perform the work of the program.

Program Statement

A program statement is a keyword in the code that does the work of the program. Visual Basic program statements create storage space for data, open files, perform calculations and do several other important tasks.

Method

A method is a special keyword that performs an action or a service for a particular program object. In code, the format for using a method is

- Object.Method Value

Where

- *Object* is the name of the object you want to change.
- *Method* is the command that will change the object.
- *Value* is an optional argument to be used by the method.

For example, this statement uses the AddItem method to put the word *Check* in the List1 list box:

- List1.AddItem "Check"

Variable

A variable is a special container that holds data temporarily in a program. You create variables to store calculation results, create filenames, process input and so on. Variables can store numbers, names and property values.

12.10 Writing NMRScripts in Visual Basic 6.0

12.10

12.10.1 Introduction

Visual Basic is a programming language, complete with its own development environment and compiler, that is capable of creating stand alone application executables. Having said that, it is quite easy to learn, has several tools that ease the programming burden and is a very powerful as a scripting language. The syntax is straightforward and the 'Visual' tools for creating windows, buttons, text boxes, etc. are excellent. There are many excellent books on Visual Basic to complement the documentation that accompanies the Visual Basic Professional and Learning Editions. A recommended book for learning Visual Basic is "Visual Basic 5 from the Ground Up" by Gary Cornell (Osborne/McGraw-Hill, 1997).

12.10.2 Visual Basic Editions

There are several Editions of Visual Basic available from Microsoft; all will provide the necessary tools for creating NTNMR Scripts. The Professional Edition contains the most complete set of tools and is recommended for advanced users. The Learning Edition is suitable for most applications. The development environment is identical for all Editions of Visual Basic and the Project files that are created are compatible making it possible to create and edit scripts in any edition.

12.10.3 General Concepts

Adding NTNMR to the "References" list

Each new Visual Basic Project must have the NTNMR reference activated. This allows the NTNMR object library to be available inside of the Visual Basic script. If the NTNMR reference is not added to the Visual Basic Project, the results will be a compilation error in Visual Basic.

> *You will know that NTNMR reference has been added to your project when you try to call an NTNMR method using the object.method syntax. Immediately after you type a valid objectname followed by a period, the Visual Basic Intellisense feature will pop-up a list of available methods for the NTNMR object type.*

Select **Project|References…** to open the References dialog box. Inactive references are list alphabetically. Activate the NTNMR references by clicking to place a check mark next to the listing for NTNMR. After you set a reference to an object library by selecting the check box next to its name, you can find a specific object and its methods and properties in the Object Browser in Visual Basic.

> *A "Project Template" can be created that has NTNMR in the reference list, commonly used code, code modules loaded, etc. Create a project, add the appropriate references, code, code modules, etc. and save the project into the VB/Templates directory with the name "NTNMR Script.vbp" (you will also need to save the form). The next time you select New Project from the File menu, your template will appear.*

"Linking" to an "Object"

Most automation commands are sent to an NTNMR data file, also referred to as an NTNMR document. OLE Automation requires that the script be linked to an object, which in most cases is an NTNMR data file.

This important concept is slightly different than other scripting languages that may be familiar to you. The result is that in most cases, the script sends commands to an NTNMR *data file,* i.e. "document" and *not* the NTNMR application itself.

In order to create a link to a document, the document (data file) must exist. This does not simply mean that there is a data file open in NTNMR. The data file that you wish to link to must have been saved to disk.

There are three methods for creating a link to an NTNMR Document.

1. Linking to a specific document path

The following lines of code are used in a Visual Basic project to create a link so that a file can be manipulated:

> Dim *objectname* as NTNMR.Document
>
> Set *objectname* = GetObject ("path_to_data")
>
> *objectname* is used here to represent any variable name that you wish to use in the Visual Basic project to refer to the data file. In all of the examples in this chapter, the variable "Data" is used.

path_to_data is used to represent the complete path to the data file. The complete path *must* be specified.

A specific example is shown below:

```
Dim Data as NTNMR.Document
Set Data = GetObject ("C:\ntnmr\data\test1.tnt")
```

> *The Dim objectname as NTNMR.Document statement is usually placed in the declarations section of the Visual Basic project. For simple applications, this is not mandatory, but it is good practice, nonetheless.*

2. Linking to a general document path

This method uses the NTNMR application to first get the path to the top most document. This allows one to create more generalized scripts that are not specific to a particular path name. This is the method that is recommended for scripts that are used to process data.

```
Dim data_objectname as NTNMR.Document
Dim app_objectname as NTNMR.Application
CreateObject ("NTNMR.Application")
Set app_objectname = GetObject(, "NTNMR.Application")
objectpath = App.GetActiveDocPath
Set data_objectname = GetObject (objectpath)
```

The advantage of this method is that the specific path to the data file does not have to be known, allowing generalized scripts to be written that will detect the path to most data file.

3. Using the provided GetActiveDoc code module

A code module is supplied to make the task of creating a link to an NTNMR data file even easier. This code module uses the application interface to coerce the current document path and then creates the object using the user supplied *objectname*.

Add the code module called "get_active_doc_path" by selecting Add Module from the Project menu in Visual Basic, clicking on the Existing tab and navigating to the modules directory in the NTNMR/Scripts folder. (**Note the location of the supplied code modules may vary depending on the installation. The code modules can be found on the NTNMR CD.)*

Once this code module is loaded, the much simpler syntax below can be used to create a link to the top most document in NTNMR.

> Dim *data_objectname* as NTNMR.Document
> Dim App as NTNMR.Application
> GetActiveDoc data_objectname

Note that in this case the object name defined in the Dim statement for the application object *must* be 'App,' as shown above. The code module also provides error checking so that if the document link cannot be created, your project will not crash.

Working on an "Object"

Once a link to an object (i.e. data file) has been created, applying NTNMR commands to the data file is straight forward. The general syntax is as follows:

> objectname.ntnmr_command

Note that this general syntax is illustrated in the preceding section where each scriptable command is discussed in detail.

A specific example is shown below:

> Data.FourierTransform

Many commands also require or have optional parameters that are specified following the command name. Several examples follow:

> Data.ExponentialMultiplication 5
> Data.SaveAs ("C:\ntnmr\data\mydata1.tnt")
> Data.SetTable "ph0", "0 1 2 3"

Very Important Note for Visual Basic Users

The syntax for using multi-parameter commands in Visual Basic requires some attention to detail. Take the examples below:
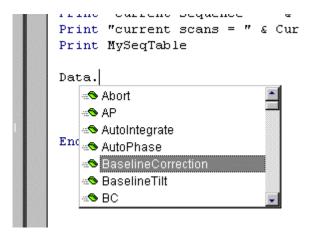
1. Data.SetSelection 200, 2500 'okay

2. Data.SetSelection (200, 2500) 'this example will fail!

At first glance, example 2 might appear to be correct. In fact, that is the form in which all commands are presented in this manual and in the IntelliSense help in Visual Basic. Visual Basic, however, will immediately return a syntax error when the line in example 2 is entered, while it will not object to the code in example 1. Virtually all commands in NTNMR return a 'boolean' (i.e. true/false) which can be used for error checking and when the parentheses are included as in example 2, the expression must be set equal to a variable. The following syntax would not generate a compile error:

3. myVariable = Data.SetSelection (200, 2500) 'okay

In fact, if the above example is written without the parentheses, Visual Basic would return an error! Finally, commands that have only one parameter will compile correctly with or without parentheses when set equal to a variable or when written as in example 1 above.

If a link to a document was created and the NTNMR References have been activated in the current project, Microsoft's IntelliSense feature will be available for NTNMR commands. This is a feature that automatically gives visual queues regarding programming syntax. IntelliSense will also recognize the Object type and pop-up a list of available properties and methods for that object. This is true for all Visual Basic Objects, including the NTNMR Object. The graphic below shows the IntelliSense pop-up for the NTNMR Object type.

Figure 12.3 IntelliSense Pop-up window

12.10.4 A Step by Step Example

> *If Visual Basic and/or NTNMR have not been installed, install each program using their respective installation programs.*

The following step-by-step procedure outlines the creation of a simply script to perform a Baseline Correction, Exponential Multiplication and a Fourier Transform on a 1D data set.

1. Open an NTNMR data file that has previously been saved.

2. Select **File|New Project…** in Visual Basic. The New Project Dialog (shown below) will show several choices depending on the edition of Visual Basic that is used. Choose "Standard EXE" and click "OK."
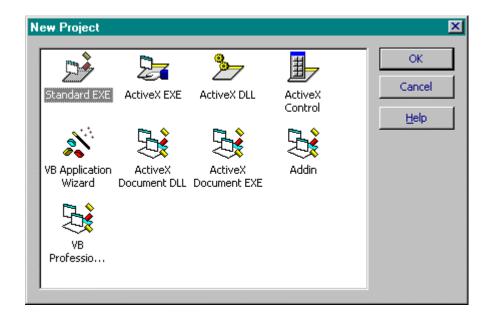
Figure 12.4 Visual Basic **New Project** Window

3. A blank project window will appear with a blank Form window visible. The figure below indicates several of the key components of the Visual Basic window.
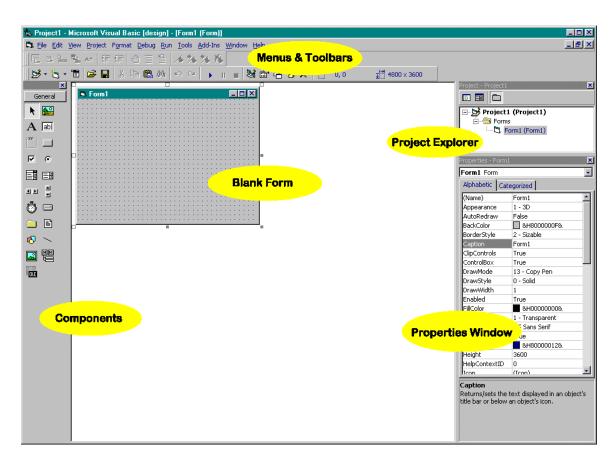
Figure 12.5 Visual Basic **Project** Window

- The Menus and Toolbars are similar to any Windows application. Only the menu items and tools that are necessary for basic script writing are discussed here.

- The Project Explorer allows navigation throughout the entire Visual Basic Project. Some advanced script may have several Forms listed in this window. There are also buttons in the upper left of the Project Explorer for toggle between the Form and Code windows.

- The Blank Form window is where buttons, text boxes, etc. are added that will be displayed when the script is run. This is the 'Visual' part of the application.

- The Properties Window will display the details about each of the buttons, text boxes, etc. that are placed on the Blank Form.

- The Components toolbar is a selection of icons representing controls that can be added to the Form window. Click once on a control to select it, and the double-click on the Form to add the selected control. New controls can be added to the toolbar.

4. Select **Project|Reference** and this will open the References dialog shown below. Scroll through the list and select **NTNMR**. You must activate the NTNMR type library for the current Visual Basic Project; this allows Visual Basic to be "aware" of the NTNMR "document". *This must be done for each new script that you create in Visual Basic.* (Note: It may be necessary to manually locate the NTNMR reference file the first time Visual Basic is run. Do this by clicking on the **Browse…** button and navigating to the NTNMR folder. Locate the "bin" folder and select the file "NTNMR.tlb")
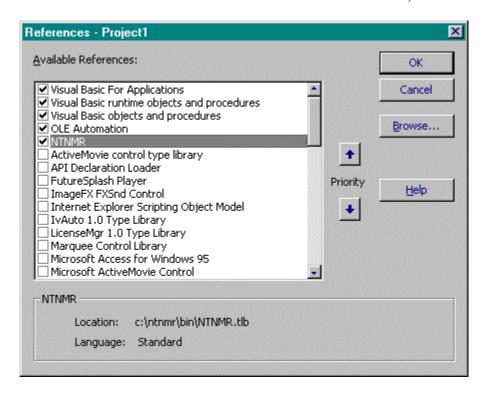
Figure 12.6 Visual Basic **Preferences** Window

5. Position the mouse pointer over the **Command Button** component on the Components Toolbar. Notice that when the mouse pointer is paused over an item on the Component Toolbar, a tool tip displays the component name.

6. Double-click on the **Command Button** component on the Components Toolbar. This will add a command button to the blank Form in the Visual Basic window. The button can be moved and re-sized on the Form.

7. Double-click in any blank spot on the **Form** window in the Visual Basic window. This will activate the **Code** window. (**Note: the Code window can be activated by clicking on the code button in the project explorer window.*)

The **Code** window has several features that are immediately important.

- The **Main Code** window: this is where the Visual Basic code (i.e. the "script") is written.

- The **Event** combo box (at the top right of the code window): this is a pop-up list of all the Events that the working Form can respond to.

- The **Controls** combo box (at the top left of the code window): this is a pop-up list of all of the Controls that are on the working Form.

8. From the **Controls** combo box select "Command1." This activates the portion of the code window that is associated with the command button that was created in step 6. (Note: The name of the button as assigned by Visual Basic can be changed. Also, the name that is used when referring to a control in Visual Basic does not have to be the same as the text name that is used to identify the control in the Form window.)

9. The following lines of code are automatically inserted into the Code Window by Visual Basic:

```
Private Sub Command1_Click()
End Sub
```

This is indicating that all codes inserted in between these two lines will be executed when the button named "Command1" is clicked.

Note that the Control "Command1" that was added to the form can respond to many events. The **Events** pop-up menu allows you to jump to the section of the code window where you would insert code for the selected Event.

10. The first step to writing the code is to insert the lines that create the link between Visual Basic and the NTNMR Data file (i.e. "Document"). This is done using the following two lines:

```
Dim Data As NTNMR.Document
Set Data = GetObject("path_to_data")
```

The Dim statement in Visual Basic is used to define a variable. The first line assigns the variable "Data" to the NTNMR document type.

The second line uses the GetObject command to assign a path name to the data file that the script should work on.

Insert these lines as shown in the example below. Be sure to type the **complete path name** to the data that you opened in NTNMR in step 1.

```
Private Sub Command1_Click()
Dim Data As NTNMR.Document
Set Data = GetObject("c:\ntnmr\data\sds-c13.tnt")
End Sub
```

11. Next add some data processing lines to the script. The following general syntax in used:
    data_file_variable_name.ntnmr_command

    Add the two lines as shown in the example below to perform a DC Baseline Correction and a Fourier Transform to the data

```
Private Sub Command1_Click()
Dim Data As NTNMR.Document
Set Data = GetObject("c:\ntnmr\data\sds-c13.tnt")
Data.BC
Data.FT
End Sub
```

Notice that when the defined variable name for the data file is typed ("Data" in this case) followed by a period, Visual Basic pops up a list of available commands that can be applied. The arrow keys on the keyboard can be used to navigate this list. *If you do not see the pop-up list of command,s go back to step 4 and verify that the NTNMR references file has a check next to it.*

12. Select **Run|Start** in Visual Basic (or press the F5 key on the keyboard) to run the script.

13. The Form that was created with a single button named "Command1" should appear. Click the "Command1" button.

14. Switch views to display the NTNMR window to check to see that the data was processed.

12.11 Example NMRScripts - Visual Basic 6.0

12.11

The scripts below are included as project files in the subfolder called "example scripts" in the script folder, or can be obtained from Tecmag Technical Support if not included on your CD-ROM. Note that in the examples below the Visual Basic comment character (') has been used to show comment lines.

12.11.1 Example 1: 1D Processing Script

Description:
This script applies simple processing to a data file. This script is mainly for demonstrating scripting syntax and is of little practical use as a script. All of the processing done in this script can be handled in a nDFT processing template.

Comments:
A form was created using two buttons and a label. The default button names used by Visual Basic were used - Command1 and Command2. The Command1 button was given the caption "Quit" and the Command2 button was given the caption "Run" by editing the Caption field in the Properties window for both. Similarly, the text label, named Label1, was given the default text string as shown in the form below.

The code is divided into three sections. The top section (Select (General) in the Controls combo box, and (Declarations) from the Events Combo box) declares the variable Data as an NTNMR object of the type Document.

The second section (Control = Command1, Event = Click) is very brief. The single line of code "End" is called when the Command1 button ("Quit" button) is clicked. This quits the script.

The third section of the script is where the data processing actually occurs in response to the "Run" (Command2) button being clicked. Notice also that the Label1 control is utilized to provide feedback to the user on the progress of the script. Just before the processing is started Label1 is set to the text "Working…" Label1 is reset again before the Auto Phase command is called. Finally, when the script is completed, Label1 is set to inform the user that processing is completed. This script also updates the text on the title bar of the window in the line "Form1.Caption … "

Form:

Figure 12.7 1D Processing Script Form.

Script:

```
'define the variable Data as an NTNMR document
Dim Data As NTNMR.Document

--------------------------------------------------------------------------------------------------------------------
----------
Private Sub Command1_Click()
'quit the script when clicking on Quit button
End
End Sub

--------------------------------------------------------------------------------------------------------------------
----------
Private Sub Command2_Click()
'execute the commands below when clicking on Run button
'define the data set to be used
Set Data = GetObject("C:/ntnmr/data/c13.tnt")
'set the title bar of the script window
Form1.Caption = "1d process"
'display some text in the script window
Label1.Caption = "Working..."
'processing...
Data.BC
Data.SBS 90, 9000, 1
Data.FT
Label1.Caption = "Working... Auto Phase"
Data.AP
'prompt the user to quit the script
Label1.Caption = "Script completed. Click Quit to end."
End Sub
```

12.11.2 Example 2: Single Click Acquisition and Processing

Description:
This script demonstrates using Visual Basic to acquire and process a data set.

Comments:
A form was created using two buttons and a label. The default button names used by Visual Basic were used - Command1 and Command2. The Command1 button was given the caption "Quit" and the Command2 button was given the caption "Run" by editing the Caption field in the Properties window for both. Similarly, the text label, named Label1, was given the default text string as shown in the form below.

The code is divided into three sections. The top section (Select (General) in the Controls combo box, and (Declarations) from the Events Combo box) declares the variable Data as an NTNMR object of the type Document.

The second section (Control = Command1, Event = Click) is very brief. The single line of code "End" is called when the Command1 button ("Quit" button) is clicked. This quits the script.

The third section of the script is where the data processing actually occurs in response to the "Run" (Command2) button being clicked. Notice also that the Label1 control is utilized to provide feedback to the user on the progress of the script. Just before the processing is started, Label1 is set to the text "Working…" Label1 is reset again before the Auto Phase command is called. Finally, when the script is completed, Label1 is set to inform the user that processing is completed. This script also updates the text on the title bar of the window in the line "Form1.Caption … "
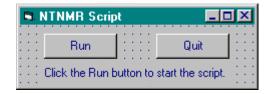
Form:

Figure 12.8 1D Acquisition and Processing Script Form.

Script:

```
'define the variable Data as an NTNMR document
```

```
Dim Data As NTNMR.Document

--------------------------------------------------------------------------------------------------------------------

Private Sub Command1_Click()

'quit the script when clicking on Quit button

End

End Sub

--------------------------------------------------------------------------------------------------------------------

Private Sub Command2_Click()

'execute the commands below when clicking on Run button

'define the data set to be used

Set Data = GetObject("C:/ntnmr/data/c13.tnt")

'set the title bar of the script window

Form1.Caption = "1d Experiment"

'display text in the script window

Label1.Caption = "Acquisition"

'start the acquisition loop

Data.ZG

Data.CheckAcquisition

Do While Not Data.CheckAcquisition

Loop

'change text in the script window

Label1.Caption = "Processing"

'processing...

Data.BC

Data.SBS 90, 9000, 1

Data.FT

Label1.Caption = "Processing... Auto Phase"

Data.AP

'prompt the user to quit the script

Label1.Caption = "Script completed. Click Quit to end."

End Sub
```

12.11.3 Example 3: A 'Batch' Script Example

Description:
Often times you may want to create a script that will simply execute a series of processing steps and then quit, without user interaction. Scripts that run in 'batch' mode, functions essentially the same as user defined commands. Note that in Visual Basic, a form *must* be created, since the code that is written is associated with its form (and not the other way around!). Even though in batch mode, no user interaction is required, the form is still of some use. The example shown below uses a label on the form to display text messages updating the user on the progress of the processing. For simple scripts, this may not be useful since the execution is likely to be very fast. For longer, more complicated scripts, however, it is a good practice to inform the user of the progress. This script also uses the provided code module 'get_active_doc_path' to create the link to the active document. Using code modules is a very important concept and can save a good deal of time in script writing.

Comments:
A Form was created and re-sized that contains only a label (called 'label1') in the example. The label is used to inform the user of the progress of the script.

The code is divided into two sections. The top section (Select (General) in the Controls combo box, and (Declarations) from the Events Combo box) declares the variable Data as an NTNMR object of the type Document.

The second section of this script, written in the Form_Initialize event, introduces two new concepts, one specific to NTNMR and one specific to Visual Basic. First, the code module 'get_active_doc_path' is used to create the link to the top most data file. This is a very convenient method of creating a link to a data file. The code module provides what is essentially a new command that can be used in the project: GetActiveDoc *objectname*. The code module must be added to your project before the script below will run!

The second new concept that is introduced is the use of the With … End With syntax. Earlier, the general syntax of *objectname.ntnmr_command* was introduced. This script shows the shortcut syntax of:

> With objectname
> .*ntnmr_command*
> .*ntnmr_command*
> .*ntnmr_command*
> End With

Form:
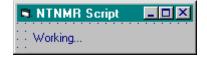
Figure 12.9 Working… Script Form.

Script:

```
'define the variable Data as an NTNMR document
Dim Data As NTNMR.Document
Dim App As NTNMR.Application

----------------------------------------------------------------------------------------------------------------
--------
Private Sub Form_Initialize()
GetActiveDoc Data
With Data
.BC
.EM
.FT
.AP
End With
'exit the script
End
End Sub
```

12.11.4 Example 4: Using "File Open" Dialogs

Description:
This example adds a directory and a file list dialog boxes so that the path to the data does not have to be hard coded into the script, as in examples 1 and 2. The dialogs used here are the standard (and somewhat clumsy!) Visual Basic dialogs. A Drive list component is also available. This example integrates the drive and file list boxes into the main form. These boxes could, however, be used in a separate form that would only be opened when the user clicked a pre-defined button. Another option illustrated later (see examples 6 & 7) is the more elegant Microsoft Common Dialog control. This control allows the more common File Open dialogs that are used throughout the NTNMR (and the Windows operating system) to be used.

Comments:

A Form was created and several components were added:

- Directory List Box -- *appears at left*
- File List Box -- *appears at right*
- Label3 -- for information only, caption "Select a directory:"
- Label4 -- for information only, caption "Select a file:"
- Label1 -- for information only, caption "Selected file:"
- Command1 --when clicked data processing is executed, caption "Process 1"
- Command2 -- when clicked data processing is executed, caption "Process 2"
- Command3 -- when clicked the script is terminated

The following sections of code were added:

In the General/ Declarations section the variable Data is defined as an NTNMR Document and three other variables are defined using the Dim statement. These other variables are defined in the General section so that they will be available to all of the Sub procedures.

Both the Command1_Click procedures are similar. They both check to see that a data file has been selected using the Directory and File boxes. This is accomplished through an If-Then statement. Note that this code also adds a message dialog if no data file has been selected. The MsgBox line illustrates the syntax for this.

The Command3_Click procedure quits the script.

The Dir1_Change() procedure sets the variable DirPath to the path to the selected folder when changed in the directory box. The path for the file selection box is also set here, "File1.Path = DirPath"

The File1_Click() procedure contains the code to get the complete path to the selected data file in the file selection box. First, the variable DirPath is assigned to the path given by the Directory box (Dir1.Path). The path to the currently selected directory is always available. Next, the variable DataFileName is assigned to the data file selected in the file selection box (File1.filename). The variable PathToData is then created from the DirPath and the DataFileName variables. Finally, the Caption for Label2 is updated to display the complete path to the selected data file.

Form:
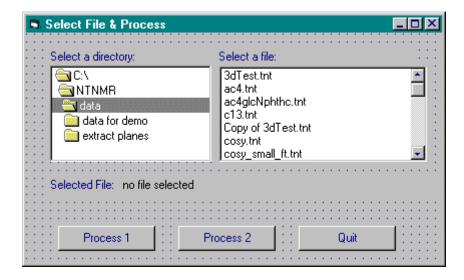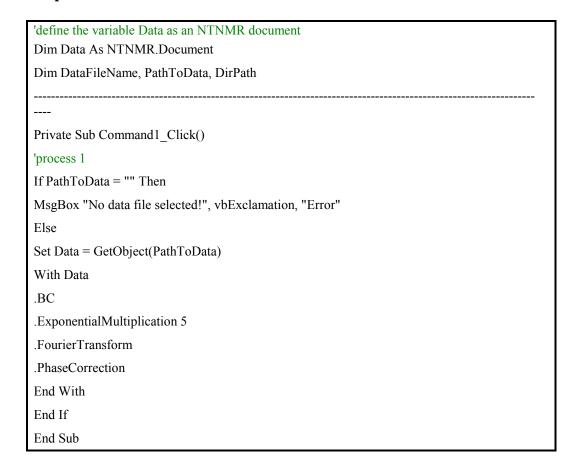
Figure 12.10 Example of **File Open** dialog

Script:

```
'define the variable Data as an NTNMR document
Dim Data As NTNMR.Document

Dim DataFileName, PathToData, DirPath

-------------------------------------------------------------------------------------------------------------------
----

Private Sub Command1_Click()

'process 1

If PathToData = "" Then

MsgBox "No data file selected!", vbExclamation, "Error"

Else

Set Data = GetObject(PathToData)

With Data

.BC

.ExponentialMultiplication 5

.FourierTransform

.PhaseCorrection

End With

End If

End Sub
```

```
--------------------------------------------------------------------------------------------------------------------
Private Sub Command2_Click()
'process 2
If PathToData = "" Then
MsgBox "No data file selected!", vbExclamation, "Error"
Else
Set Data = GetObject(PathToData)
With Data
.BC
.ExponentialMultiplication 10
.FourierTransform
.AP
End With
End If
End Sub

--------------------------------------------------------------------------------------------------------------------
Private Sub Command3_Click()
End
End Sub

--------------------------------------------------------------------------------------------------------------------
Private Sub File1_Click()
DirPath = Dir1.Path
DataFileName = File1.filename
PathToData = DirPath & "\" & DataFileName
Label2.Caption = PathToData
End Sub
```

12.11.5 Examples 5 & 6: Experiment Scheduler & Kinetics Experiment

These two examples are somewhat complicated and intended for the advanced user. They are made available in an *.exe form so that they can be used immediately. The complete project files are also available. Since these examples are fairly long, the code is not shown here. The Forms are shown for reference

and a brief description of the script is given. The complete code can be examined by opening the project file in Visual Basic.

Experiment Scheduler Description:
This example illustrates using Visual Basic to queue several experiments in a row, consecutively. The script allows the user to choose several pre-defined experiments (saved as data files), and queue them to be run. Use the Add button to add a new experiment to the list. The code for the Add button shows an example of using the more elegant Microsoft Common Dialog for file/open operations.
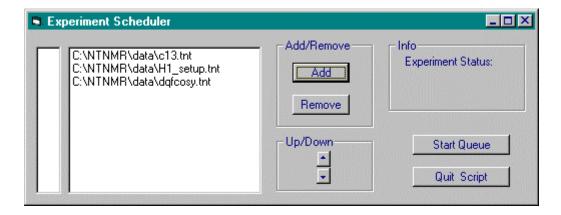
Experiment Scheduler Form:

Figure 12.11 Experiment scheduler dialog box

Kinetics Experiment Description:

The Kinetics experiment works with the open and top most data file to acquire 'Number of Exps.' experiments with the time in between experiments specified by the parameter 'Wait Time' in seconds. The individual data files are saved to a disk in the format *data_file_name.experiment_number*.tnt. Note the use of the timer function for creating the wait time in between experiments.

Kinetics Experiment Form:

Figure 12.12 Kinetic Script dialog box.

12.12 Script List

12.12

A number of scripts are included with the standard NTNMR software distribution. These scripts serve both to enhance the function of NTNMR, and to demonstrate the possibilities and techniques of scripting. The *.vbs scripts can also be used as convenient starting points for user-defined scripts. The following alphabetical list indicates the use of many of the scripts included in this set.

Available Scripts		
Name	**Type**	**Description**
1D bc, em, ft, ap	(Processing)	Automatically apply baseline correction, Exponential multiplication, fourier transform and automatic phase correction. For exponential multiplication, the value of LB must be preset in the dashboard.
2Dbaseline correct	(Processing)	Baseline correction for a 2D data set. All records are corrected by the same amount, rather than being individually corrected.
2D Shearing	(Top Level)	Shifts the each record by a small amount to render a skewed data set

		rectangular. Each row is shifted by "nx" where n is the row number –1 and "x" is provided by the user.
2D to 1D	(Processing)	Converts a 2D dataset to a 1D dataset, where each record from the second dimension is placed at the end of the previous one. Useful for displaying results of nutation or relaxation experiments on a single screen.
2D TPPI Processing 2	(Processing)	Processes 2D TPPI format data set (e.g. COSY, HETCOR).
AddSubstract (single record	(Top Level)	Adds two 1D data sets together.
AddSubstract 2D	(Top Level)	Adds two 2D data sets together
AddSubstract	(Processing)	Starts an interactive menu that prompts the user to choose two 1D data files to be added or subtracted. The result is placed in the first dataset opened.
ApolloAG	(Hardware)	Automatic receiver gain setting for Apollo.
AutoProcess 1D	(Processing)	Performs a baseline correction, Exponential multiplication, fourier transform, Automatic phase correction and fits the data to the window. Useful as a starting point for user processing.
AutoProcEcho	(Processing)	Same as above for echo data.
AutoShim (axials)	(Top Level)	Starts an automated shimming sequence for axial shims ('spinning' shims) based on some pre-defined parameter settings. Edit the script to change the parameters.
AutoShim (complete – no spin)	(Top Level)	Starts an automated shimming sequence for all shim terms, for non spinning sample, based on some pre-defined parameter settings
AutoShim (complete – spin)	(Top Level)	Starts an automated complete shimming sequence for all terms, using sample spinning for axial terms, based on some pre-defined

		parameter settings.
Bruker AG	(Hardware)	Automatic receiver gain setting for DSPect.
Calculate SNR	(Processing)	Starts the Signal To Noise calculation dialog. The user is prompted to enter the boundary of a region with noise only and the position of the largest peak. The positions can also be determined interactively using the cursor.
Chirp Generator	(Misc.)	Demonstrates creating a sequence shape table, in this case a quadratic phase ramp for a 'chirp' pulse. User is prompted for experimental parameters.
Command Line	(Top Level)	Opens the command line window and allows the user to conbtrol NTNMR with 2 or 3-character command words.
cpmg proc setup	(Processing)	Converts 1D data obtained using a CPMG sequence and to a 2D data set for processing.
Dashboard customize	(Top Level)	Opens the dashboard customize window. See chapter 4 (dashboard) for details.
DSPect Homo Decoupling	(Acquisition)	Sets up a Homo-nuclear decoupling experiment with the DSPect. The script's usage is described in detail in the DSPect's user manual.
FIDMath	(Processing)	Opens the FIDMath Dialog. This script allows the user to perform some mathematical operations on the data. For example: multiply by a constant, change the data from real/imag. to magnitude/phase, compute statistics on a selected range of data points, etc.
Filtertest	(Misc.)	Runs the Filtertest experiment for hardware testing.
Find Biggest Peak	(Processing)	Find the position of the largest peak and gives the amplitude of the real

		part of the spectrum at that point.
Kinetics Exp	(Acquisition)	Opens the Kinetics Experiment dialog window. The user can choose how many times to run the selected experiment, and the interval between runs. Each file will be saved with a different name. This script is described in detail in the section on VB programming, above.
MS Processing and Display	(Processing)	Process and diplays multi slice imaging data.
MS Processing	(Processing)	Processes multi slice imaging data.
MS Processing_two receiver	(Processing)	Processes multi slice imaging data acquired with two receivers (interleaved).
MS Processing_two receiver_planes	(Processing)	Processes multi slice imaging data acquired with two receivers.
MS Transpose	(Processing)	Transposes Multi Slice data from the acquisition format to the conventional format (slice number becomes the 3rd dimension)..
Multi point to single point	(Processing)	Rearrange the data from a 1D dataset to a 2D data set. Each record has a dimension determined by the user. Used for processing some pseudo-single-point acquisition experiments.
NMRScriptEditor	(Top Level)	Starts the NMR Script editor described earlier in this chapter.
NTNMR Default paths	(Misc.)	Queries the system for the present setting of the NTNMR paths. Useful primarily as a demonstration of how to get path information for creating or opening files.
Online Manual	(Top Level)	Brings up the NTNMR online manual.
Param Set Example	(Misc.)	Example script which sets the 90, 180 pulse width, the sweep width and the number of points.

Process 2D (Magnitude)	(Processing)	Processes a 2D experiment (e.g. COSY) in Magnitude mode.
Process and Display MS	(Processing)	Processes Multi slice data set.
SaveNTNMRPrefs	(Misc.)	Save NTNMR registry settings in a file (ntnmr.reg). This is useful for backing up or transferring settings from one user or computer to another. Double-clicking the file in Explorer will load the settings to the current user's registry.
Scale Data	(Processing)	Scales the data by a value entered in the dialog box.
Scale Lock	(Hardware)	Gets the lock level
Set Dec Freq	(Acquisition)	Sets the decoupler frequency according to the position of the cursor on the spectrum.
Set New SW	(Acquisition)	Sets the spectral width according to the highlighted area of the data. Before running the script, highlight the region of interest in the spectrum. The width of the region selected will become the spectral width and the spectrometer frequency will become the center of this region.
Set rd and ad	(Configuration)	Sets the values of rd and ad relative to the SW and the number of points. The sequence must have these variables for the script to work for DSPect.
Table Wizard	(Configuration)	Starts the table wizard. See chapter 10 (sequence editor) for details.
Update Parameters	(Configuration)	Updates the global parameters file
ZG Save	(Acquisition)	Starts the experiment that is in focus in the NTNMR window and saves it when it is done. While the script is running, a small window display the status of the experiment.

12.13 Frequently Asked Questions

12.13

Why can't I use a script to process the data that I just acquired?

This is probably because the window that was used to acquire the data was first created by selecting **File|New**. Visual Basic needs a file path to create a link to a document. If the data window was created from the File menu by selecting New, then there is no path since the data has never been written to disk. The solution to this problem is to acquire the data in a window that has previously been saved. When a zero-go command is executed, the data window will be renamed in NTNMR. However, Visual Basic will still be able to create a link to the file.

Can I use a language other than Visual Basic 6.0 to write "scripts?"

Yes - however, Visual Basic 6.0 is the only automation language that is officially supported by Tecmag. This means that languages such as Java, C++, WinBatch, Python, etc. can be used, but Tecmag cannot guarantee functionality and the support, if any. The support Tecmag can offer for languages other than VB will be limited.

References

Numerous books on Visual Basic are available at both general bookstores and computer discount houses.

Internet references (current as of July 2002).

VBScript language reference:
http://www.microsoft.com/scripting (choose *documentation*, then *script technologies*, and finally *VBScript*.

VBScript Tutorial:
http://msdn.microsoft.com/library/en-us/script56/html/vbstutor.asp

Scripting examples:
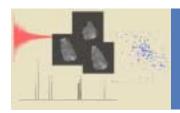http://msdn.microsoft.com/library/en-us/script56/html/wsconWSHWalkthroughs.asp

File system
http://msdn.microsoft.com/library/en-us/script56/html/sgFileSystemObjectModel.asp

Miscellaneous Links
http://msdn.microsoft.com/library/en-us/dnwinpro00/html/WindowsScriptingHost.asp
http://msdn.microsoft.com/library/en-us/dnmind99/html/cutting0999.asp

Appendix A

A.1 NTNMR File Types

A.1

- System Files(".dll", ".exe", ".seq", ".ini", ".con", ".aff", ".seq", ".txt",) see below for more details.

- Data Files (*.tnt) see below for more details

- Sequence Files (*.tps) see below for more details.

- Shim Files (*.shm) see below for more details.

- NDFT Templates (*.tft) see below for more details.

- Parameter File (*.txt) see below for more details.

- Print Templates (*.prt) see below for more details.

- Print Template Parameter File (*.par) see below for more details.

- Visual Basic Script (*.tps) see below for more details.

- Visual Basic Executeable (*.exe) see below for more details.

A.2 System Files

A.2
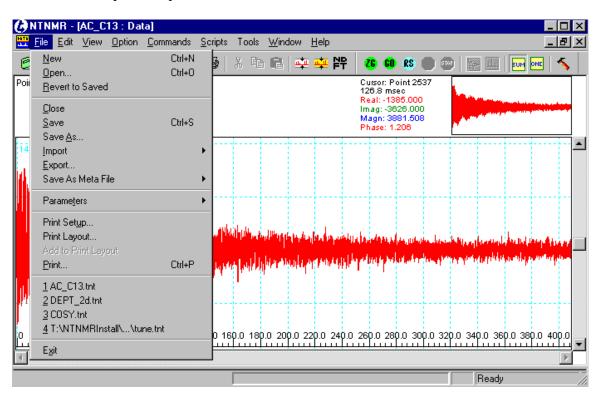
System files can be categorized as all files that are associated with the initialization and running of the Tecmag system. These files can be found in the bin, config, system and temp directories of NTNMR. These files should not be changed once the system is installed and any changes in these files will affect the operation of NTNMR.

A.3

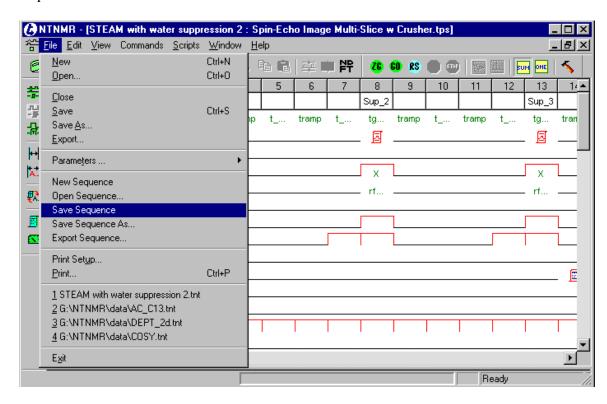# A.3 Parameter and Data files

Parameter and Data files are files that can be saved while using NTNMR. Data files end in the ".tnt" extension. They contain the acquired data as well as the dashboard and pulse sequence information.
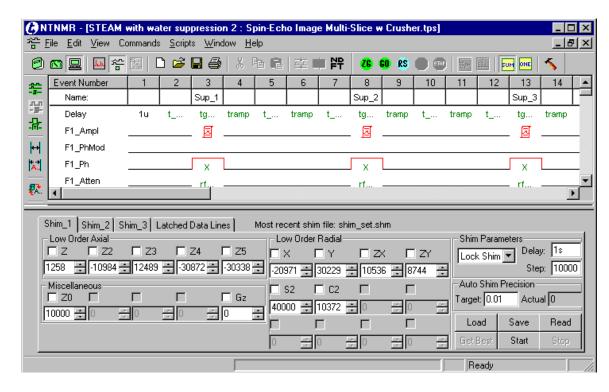
A.4 | A.4 Sequence Files

Sequence files have the ".tps" extension. They contain only pulse sequence and sequence variable information.
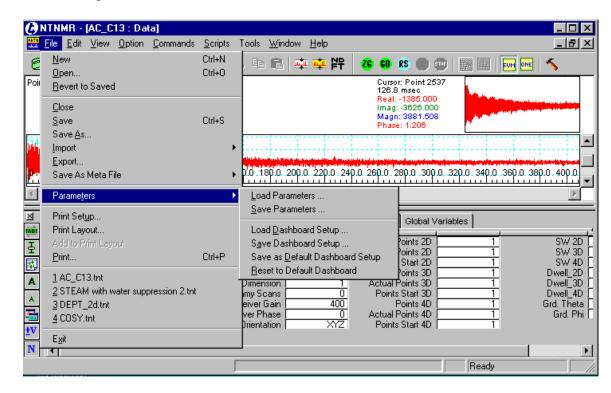
A.5 Shim Files

Shim file are saved with the ".shm" extension. They contain both the values of the shims as well as the shim configuration.
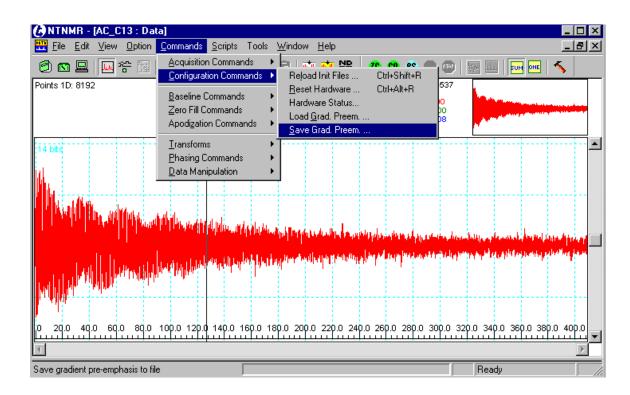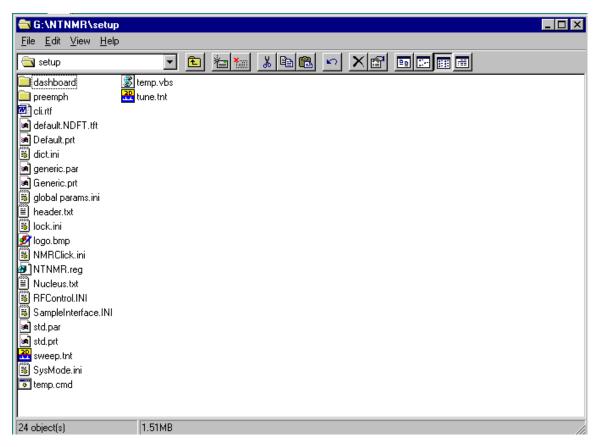
<table>
<tr><td>**A.6**</td></tr>
</table>

A.6 Dashboard parameters

Dashboard parameters can be saved into a file with the ".txt" extension.
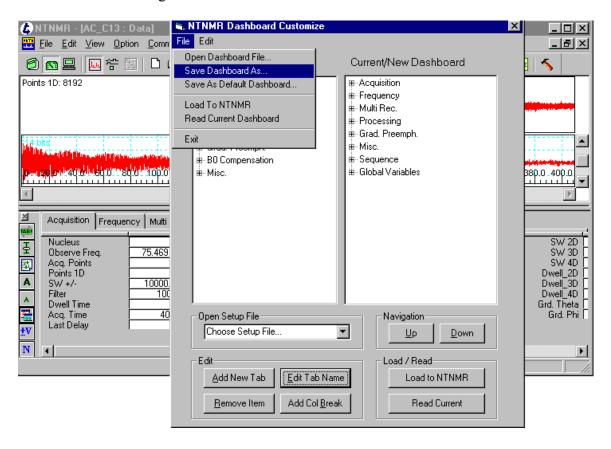

Specialized dashboard parameters such as the preemphasis and Bo compensation values can be saved separately. They get saved to a file with a ".txt" extension also but have there on directory in the setup folder. The default file that gets loaded when NTNMR is started is called "Preemph.txt."
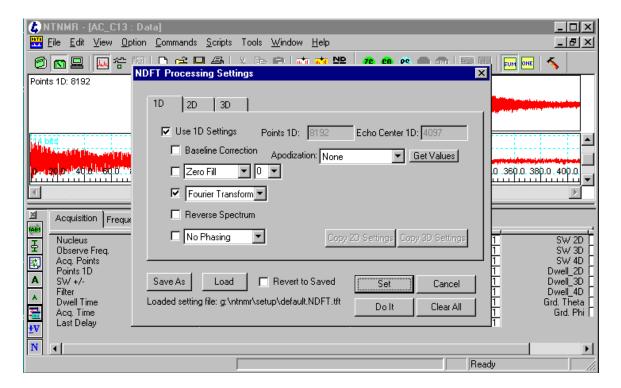
Dashboard settings are saved in the Setup/Dashboard folder. It contains a list of what parameters are displayed in the Dashboard. When NTNMR starts it loads the dashboard settings saved in the default.txt file.
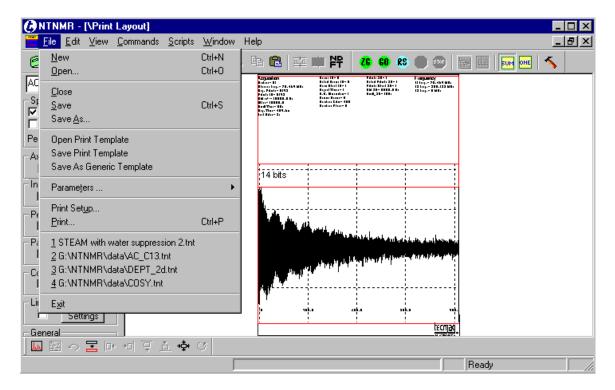
A.7

A.7 NDFT template Files

NDFT template files contain saved processing parameters. These files are saved with the ".tft" extension.

A.8

A.8 Print Template Files

Print Template files are saved with the ".prt" extension. The generic print template is saved in the setup folder as "generic.prt." It contains saved print template settings. Associated with these print template files are ".par" which contain the parameter list.

A.9 Visual Basic Scripts and executables

A.9

Visual basic scripts and executables have the ".vbs" and ".exe" extension. Visual basic executeables are made using the Visual Basic software. In order for changes to be made in the executable you will need to ask us for the source code for that script. Visual Basic scripts are made via the NMRScriptEditor or can be made also with any text editor.