

# WaveMaker

- Pulse Shaping Software  
User Manual  
Version 013



Copyright © by Bruker Corporation

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means without the prior consent of the publisher. Product names used are trademarks or registered trademarks of their respective holders.

© April 26, 2021 Bruker Corporation

Document Number: 1000058073

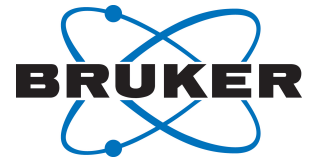
P/N: H164492

---



## Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>WaveMaker in a Nutshell</b>	<b>5</b>
2.1	Waveform Definition in Pulse Programs	5
2.2	Shape Functions	5
2.3	Shape Specific Parameters	6
2.4	Gradient Pulses	7
2.5	User defined RF Calibration Parameters	7
2.6	Making Shapes	7
2.7	Internal Settings	8
2.8	ZGOPTNS ignored	9
<b>3</b>	<b>WaveMaker Applications</b>	<b>9</b>
3.1	WaveMaker Syntax in Detail	9
3.2	Absolute vs. Relative Offsets	10
3.3	CPD Programs	10
3.4	Creating Waveforms for Existing (Standard) CPD Programs - Bi-level Decoupling	11
3.5	Gradient Waveforms	12
3.6	Adiabatic Pulses	13
3.7	J-Compensated Adiabatic Pulses	13
3.8	Composite Adiabatic Pulses	15
3.9	Adiabatic pulses in <sup>19</sup> F NMR	16
3.10	Notch Pulses	18
3.11	Optimized Notch Pulses (NEW)	18
3.12	Time Shifted Pulses (NEW)	19
3.13	Shape Definition Files	20
3.14	Using TopSpin Parameters	20
3.15	Re-defining the Phase Alignment (PA)	21
3.16	Re-defining the Excitation Bandwidth	21
3.17	Multi-frequency Waveforms	21
3.18	Shape Arrays	22
3.19	Compensation of Bloch-Siegert Interferences	23
3.20	Using Peak Lists and Hadamard Encoding Experiments	24
3.21	Multiple (Solvent) Peak Suppression	27
3.22	Adiabatic Homo-decoupling	28
3.23	Specifying the Required B1max, B1rms and B1av	29
3.24	Off-resonance Spin Locks	29
3.25	Composite Spinlocks	31
3.26	Spinlocks for Cross-Polarization	33
3.27	WaveMaker Parameter File	35
<b>4</b>	<b>Appendix</b>	<b>35</b>
	<b>List of Figures</b>	<b>41</b>
	<b>List of Tables</b>	<b>43</b>



5 Contact ..... 43

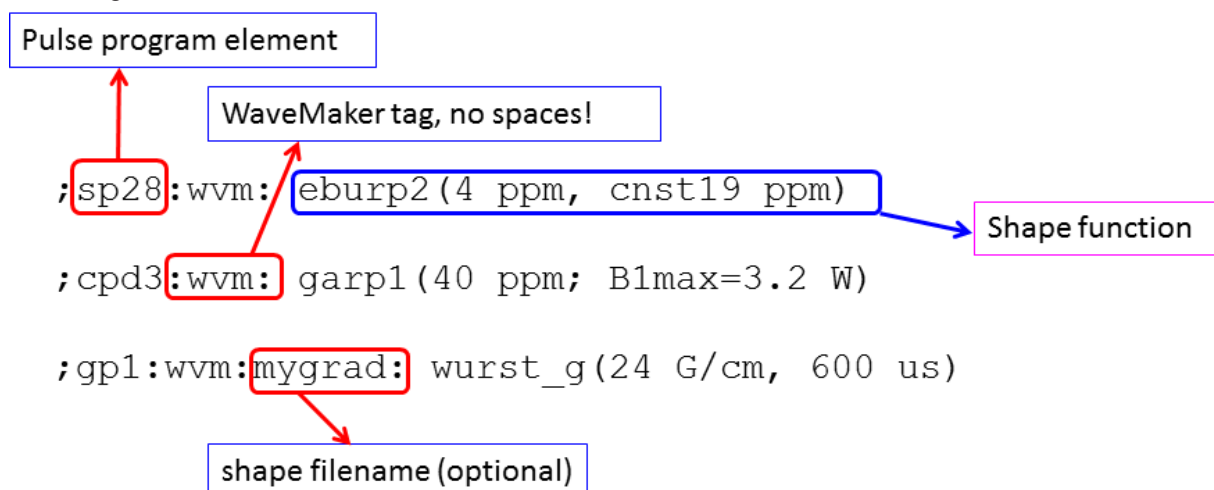
## 1 Introduction

WaveMaker is a software solution for pulse shaping that is based on waveform definitions within the pulse programs. This avoids any need to modify the PROSOL tables when new or non-standard shaped pulses and decoupling or mixing waveforms are required. It also adds flexibility to and improves portability of many standard and user developed experiments and pulse programs. Porting the WaveMaker based pulse programs to NMR systems of different field strength is straightforward and typically requires no further adjustments and PROSOL updates.

## 2 WaveMaker in a Nutshell

### 2.1 Waveform Definition in Pulse Programs

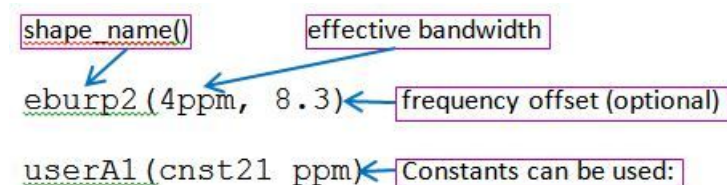
Pulse shapes are defined as shown below either in the comment section of the pulse programs or using userA parameters or both. The shaped pulses are then created from the TopSpin command line by executing the **wvm** command.



Note that pulse shapes defined in the parameter sets via the userA parameters will take precedence over the definitions in the comment section of the pulse program.

### 2.2 Shape Functions

The shape functions are defined as shape name with any or all of the following four arguments in parenthesis – (1) desired excitation bandwidth (bw), (2) pulse duration (pw, optional), (3) offset (ofs, optional) and (4) phase (ph, optional) in that particular order (!).

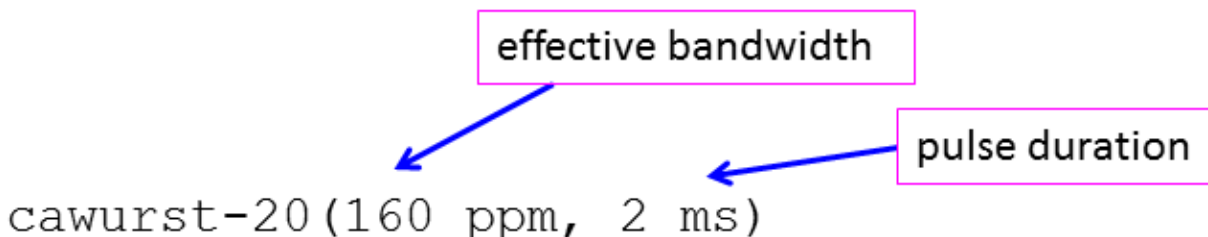




This is also the most logical order because the bandwidth is usually of most interest while the pulse duration is typically calculated from the specified bandwidth unless it needs to be set exactly, for instance in water suppression sequences, spin locks or adiabatic pulses. Finally, the offset is often specified explicitly in the pulse programs and phase is also set directly, except if several shapes need to be combined, for instance in Hadamard encoding.

Note that adiabatic pulses require definition of both, the active bandwidth and the pulse duration:

The shape function can also be defined via the userA1-userA5 parameters. For instance, eburp2(const21 Hz) shape can be defined as userA1(const21 Hz), with userA1 parameter set to eburp2. A simple change of the userA1 parameter then allows creating the same shaped pulse or decoupling waveform, but with a different shape. This provides more flexibility for the user control of the experiment setup.



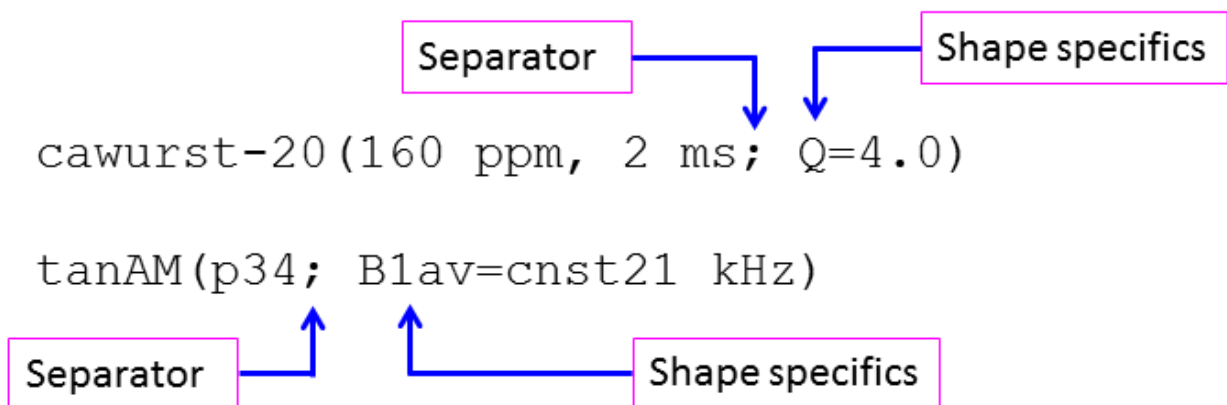
If only one of the two parameters is specified, the other will be calculated from a pre-set constant (BW), which is a solution, but not necessarily the most optimal solution.

The arguments are followed by units (us, ms, ppm, Hz, kHz, deg, etc.) and must be separated by commas. There is some freedom with the syntax and the following formats are accepted – 160 ppm, 160 p, 160ppm, 160p. The same is true for other units.

Finally, the shape names can be found in the wavemaker/shapes folders. The most frequently used shapes are listed in the [Appendix \[ 35\]](#).

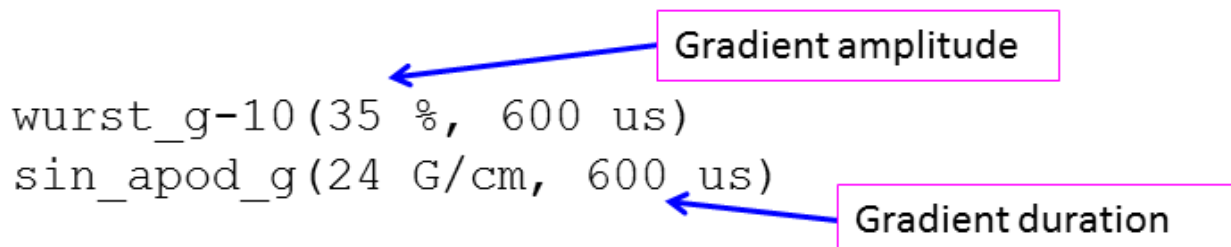
### 2.3 Shape Specific Parameters

The shape arguments (bw, pw, ofs, ph) may be followed by optional shape specific parameters. The latter must be separated from the shape arguments by a semicolon:



## 2.4 Gradient Pulses

The gradient pulses are described by gradient amplitude and the pulse duration. The gradient amplitude is given in percent of the maximum available gradient strength. Alternatively the gradient strength can be given in G/cm. In the latter case the amplitude of shaped gradients is set such that the area of the gradient is independent of the shape.


  
`wurst_g-10 (35 %, 600 us)`
  
`sin_apod_g(24 G/cm, 600 us)`

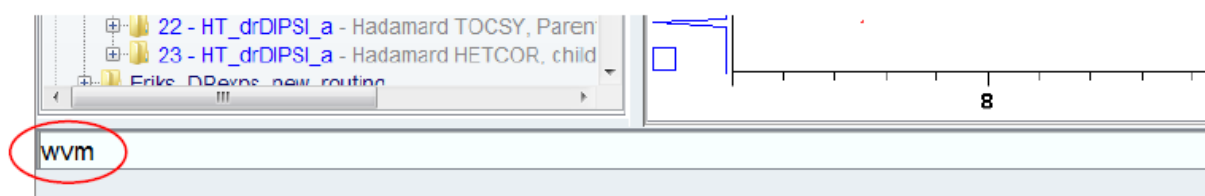
## 2.5 User defined RF Calibration Parameters

By default the WaveMaker assumes the following channel assignments:

(p1,pl1):f1, (p3,pl2):f2, (p21, pl3):f3, (p23,pl4):f4. The users can define their own assignments, for instance: reburp(25 ppm, 56 ppm) p90=p32, pl=pl32

## 2.6 Making Shapes

The waveforms are generated from the TopSpin Command line by executing the **wvm** command:



The WaveMaker will check the *pulprog* parameter and parse the corresponding pulse sequence for WaveMaker input data. The *acqu* file is parsed to determine the parameters of the RF channels. Therefore it is recommended to use the **getprosol** command to set up the correct hard pulse durations and the corresponding power levels. The WaveMaker then generates the input file, *wvm.par* and a log file, *wvm.log* in the current experiment for reference and de-bugging purposes. The generated shape files are stored in the corresponding TopSpin directories - *wave/user*, *cpd/user* and *gp/user*. The '**a**' option will set the WaveMaker calculated parameters into your current parameter set.

A plain **wvm** command creates all the necessary shape files. In the output TopSpin shows the parameters – shape names, pulse durations, power levels, etc., that are relevant for the current experiment, but does not alter any of the (*acqu*) parameters.

```

L3 Experiment Setup:
L4 p11 = 3230.5000 us
L5 spnam1 = xxx_sp1.wv
L6 spw1 = 0.0193455 W
L7
L8 Wavemaker: manual-setup requested

```

The following arguments to the **wvm** command are accepted:

- **x** – ‘dry’ run; all calculations are done as usual, except no shape files are created/saved;
- **a** – auto setup; all shapes are calculated and the shape parameters are set into the current experiment; by default this is a ‘silent’ option and no output is shown to allow for using in automation;
- **l** – shows the `wvm.log` file of the last run of the `wvm` program.
- **al** – combines the `-a` and `-l` options.
- **p** – shows the `wvm.par` file of the last run of the `wvm` program.
- **q** – the ‘quiet’ option; for use in automation; works in the same way as the auto-setup option, `-a`, except it does not activate the `asad` command making it compliant with the automation requirements.
- **b** - shows the excitation profile of the created shape(s); the `-b` option starts the Bloch simulator, `nrmrsim`; in applications with several shaped pulses the shape number can be provided, e.g. `-b 3`; the default shape number is 1.
- **i shapename** – shows the information about individual shapes stored in the `wavemaker/shapes` directory.
- **f filename** – the shape definitions are read from a file stored in the `wavemaker/par` directory rather than from the pulse program.

The `wvm.log` file stores more details about the individual shape files such as number of steps, step size, maximum phase increment, B1max (in Hz) and reference RF parameters that were used for calibrations that can be displayed using the **wvm -l** (letter ‘l’ for ‘log’) command.

Finally, the WaveMaker generated RF waveforms are appended with the `.wv` extension, the gradient waveforms with `.gp` extension and the CPD programs with `.cpd` extension to allow easy recognition and manipulation.

## 2.7 Internal Settings

---

The default clock rate in WaveMaker has been set internally to 10 MHz. This rounds off the pulse durations and step size to 0.1 us. The state-of-the-art Bruker consoles have at least 80 MHz clock rate, which allows executing the pulse shape steps with 0.0125 us accuracy. However, the pulse duration is often rounded off to 0.1 us or even coarser thus introducing round-off errors which may have adverse effects on the performance of experiments in certain cases. For that reason the default WaveMaker clock rate has been set internally to 10 MHz. The users can adjust the WaveMaker clock rate as required by adding the following entry to the `wvm_globals` file in the `TS/exp/stan/nmr/wavemaker` directory, e.g.:  
clockrate = 80 MHz or similar, depending on the user needs and console capabilities.



## 2.8 ZGOPTNS ignored

Currently WaveMaker ignores the `#ifdef ... #else ... #endif` constructs in pulse programs. This is done to avoid an easy mistake of starting the experiment with a wrong shape after the ZGOPTNS parameter was changed but re-creating shapes was overlooked. Arguably a better solution would be to create all the required shapes independently of the setting of ZGOPTNS parameter, which is the recommended solution in this version of the WaveMaker.

## 3 WaveMaker Applications

### 3.1 WaveMaker Syntax in Detail

All shapes that are necessary for the current experiment can be conveniently defined in the comment section of the pulse programs and generated by executing the **wvm** command. As discussed above, the following syntax can be used:

```
;sp13:wvm: sinc90(1ms) np=500;  
;sp23:wvm: reburp(25 ppm, cnst22 ppm)
```

Alternatively, the `userA1 – userA5` parameters can be used to hold complete WaveMaker waveform definitions avoiding the need to make changes in the pulse program comments area. The same syntax for shape definitions apply, except for the leading comment token, `;`

```
userA1 sp13:wvm: sinc90(1ms) np=500  
userA2 sp23:wvm: reburp(25 ppm, cnst22 ppm)
```

In that way automatic pulse shaping can be applied to the sequences in the standard Bruker pulse program library. Note also that waveform definitions in the parameter sets take precedence over those in the comment section of the pulse program. This allows redefining shaped pulses without the need to change the pulse program and avoids a potential confusion if the pulse program is changed at a later stage.

Here the `sp13` and `sp23` are shaped pulses as used in the pulse program and **:wvm:** (no spaces!) is the keyword recognized by the WaveMaker. It indicates that the following text is used as the WaveMaker input information. In the above examples the file names of the output waveforms are generated automatically according to the scheme: `pulprogrname_sp#`. However, users can specify the required file names as in the case of CPD programs below:

```
;cpds1:wvm:waltzH1: waltz16(20 ppm; B1max=4 kHz)  
;cpds2(sp23):wvm:sedCO: seduce1(20 ppm, 172 ppm; AM=cos) p90=p2  
;cpd3:wvm:waltzN15: waltz16(20 ppm; B1max=4 kHz) p90=p3
```

Note that longer entries can be split for better readability, for instance a shape with two excitation bands may be defined as follows:

```
;sp17:wvm: iburp1(30p,176p) iburp1(30p,-55p) np=2000
```

or alternatively in an expanded syntax as follows:

```
;sp17:wvm: iburp1(30 ppm, 176 ppm)  
;wvm: iburp1(30 ppm, -55 ppm)  
;wvm: np = 2000
```

The WaveMaker will use the information available from the current experiment to determine the type of the waveform, RF calibration data and the base frequency of the particular RF channel. In case of ambiguities or if the power level is pre-set using a dedicated power level parameter (e.g. inside a CPD program) the user can specify the power level parameter in parenthesis, e.g.

```
;cpds2(p113):wvm: seduce1(20 ppm, 172 ppm; AM=cos) p90=p2
```

In this case *plw13* will be set to the calculated power level. Note that in the above example the user nominated *p2* (rather than the default *p3*) is used along with *p/2* as a reference 90 degree pulse for the RF shape calibration purposes.

Note that shaped pulses that are used in the CPD programs are usually not present in the pulse programs and thus are 'invisible' to the WaveMaker. However, the user can nominate which shaped pulse should be used in the CPD program, for instance:

```
cpds2(sp15):wvm: cawurst_hd-8(30 ppm, 5 ms)
```

In this case *pcpd2* will be used as pulse duration. The user can also nominate the desired pulse duration parameter to be used inside the CPD program:

```
cpds2(p63):wvm: cawurst_hd-8(30 ppm, 5 ms)
cpds2(p63,sp15):wvm:codec: cawurst_hd-8(30 ppm, 5 ms)
cpds2(p63,sp15,pl63):wvm:codec.cpd: cawurst_hd-8(30 ppm, 5 ms)
```

The next three lines show further examples of creating the shaped gradient pulses and waveforms:

```
;gp1:wvm:sinc50: sinc_g(10 G/cm, 1.0 ms; np=100)
;gp7:wvm:sinc50: sinc_g(0.6m, 15G; np=60)
;gp6:wvm:sinc50: sinc_g(20 %, 1.0 ms)
```

Note that the gradient amplitude can be defined in either % (of then maximum available gradient amplitude) or in G/cm. In the case of gradient shapes the order in which the Gradient duration and amplitude are defined within the shape function is not important, as long as the units of these parameters are clearly specified. The shorthand notation of the units, such as G (G/cm) or m (ms) is also accepted with or without space. As in the case of RF pulses the parameters must be separated by comma and the shape file specific parameters within the shape function, such as the number of points (np) must be separated from the shape specific parameters by a s semicolon;

## 3.2 Absolute vs. Relative Offsets

Typically the frequency offsets for all RF waveforms are calculated as 'absolute' frequencies from the reference position at 0.0 ppm. For instance, a shape definition such as *eburp1* (200 Hz, 2.3 ppm) will create an excitation waveform with bandwidth of 200 Hz and applied at 2.3 ppm with respect to the reference frequency (0.0 ppm). Unless defined otherwise the carrier frequency is obtained from the current parameters and the relative frequency with respect to the carrier frequency is calculated and applied to the waveform. The same is true, if the required offset frequency is given in Hz or kHz, e.g. *eburp1* (200 Hz, -1.5 kHz), which on a 500 MHz system is -3.0 ppm. However, occasionally the user may need to set the relative frequency with respect to the carrier (rather than reference) frequency. This can be achieved by setting the offset parameter to zero (ppm or Hz). The shape definition then becomes *eburp1* (200 Hz, -1.5 kHz) ofs = 0.0 ppm. Now the shaped pulse will be applied at -1.5 kHz from the carrier frequency, irrespectively of the setting of the *o1p* parameter (or similar).

## 3.3 CPD Programs

The CPD (composite pulse decoupling) waveforms are created in a very similar way to the common waveforms. There are two main groups of decoupling waveforms – composite pulse (pulse width modulated, constant amplitude) CPD sequences and CPD programs based on shaped pulses that are phase cycled either inside the waveform (a single shape CPD program) or inside the CPD program. The former (single shape CPD) is implemented by default and the latter format (explicit CPD program) can be enforced by adding the *.cpd* extension to the shape filename, e.g.

```
;cpd3:wvm:wuN15: cawurst_d-2(40 ppm, 2 ms)
;cpd3:wvm:wuN15.cpd: cawurst_d-2(40 ppm, 2 ms)
```

All CPD program output filenames are appended with the .cpd extension and typically are stored in the topspin/exp/stan/nmr/lists/cpd/user directory.

### 3.4 Creating Waveforms for Existing (Standard) CPD Programs - Bi-level Decoupling

The WaveMaker checks the pulse programs for the presence of the requested pulse sequence elements (shaped pulses, CPD programs, etc.) and will refuse to create shapes that are not present in the pulse program for a simple reason – the channel information of such shapes is missing. This may be a problem if user wants to create shapes that are used by the standard TopSpin CPD programs. The solution is to indicate the channel for which the shapes are created, as shown in the example for explicit CPD program below:

```
;sp31(p63):wvm:wuCdec:f2 cawurst-2(160 ppm, 1.2 ms; Q=2.5)
;sp31(pcpd2):wvm:wuCdec:f2 cawurst-2(160 ppm, 1.2 ms; Q=2.5)
```

Note that the above examples for explicit CPD programs use shapes from the inversion or (or refocussing) directory and in case of adiabatic decoupling the adiabaticity factor should be reduced as appropriate for decoupling, typically  $Q = 1.0$  to  $3.0$  depending on the acceptable power level and decoupling sideband intensities. If the decoupling power is preset externally (i.e. in the pulse program) the power parameter can also be nominated as shown below:

```
;sp31(p63,p163):wvm:wuCdec:f2 cawurst-2(160 ppm, 1.2 ms; Q=2.5)
```

In this example the required power level and pulse duration as used inside the CPD program are nominated within the parenthesis following the shaped pulse element, *sp31*. This enables automatic setting of these parameters using the **wvm** command (**-a** option). The nominated pulse sequence elements may or may not be present in the pulse program and no checks are carried out of any CPD programs in this revision of the WaveMaker.

In case of a single shape (rather than explicit) CPD programs the required supercycle should be defined, for instance:

```
;sp31(pcpd,p163):wvm:codec:f2 cawurst-8(30 ppm, 5 ms; Q=1.3, sucyc=[p5]m4)
```

Alternatively, shapes in the decoupling (or spin lock) directory may be used, for instance:

```
;sp31(pcpd,p163):wvm:codec:f2 cawurst_hd-8(30 ppm, 5 ms)
```

Such shapes are created with the required super cycle and adiabaticity, as defined in the corresponding shape definition files.

For the standard pulse programs the decoupling can be set up using the userA1 - userA5 parameters thus avoiding the need to modify the pulse programs. For instance,

**cpdprg2: p5m4sp180**

```
userA1 sp15(pcpd, p112):wvm:CHdec:f2 cawurst(150 ppm, 1.4 ms; Q=3, L2H)
```

**cpdprg2: p5m4sp180.p31**

```
userA1 sp15(p31, p112):wvm:CHdec:f2 cawurst(150 ppm, 1.4 ms; Q=3, L2H)
```

The required decoupling power level for these two examples was 3.5 dB below the standard **prosol** power level required for GARP decoupling.

## Bi-level Decoupling

While the adiabatic decoupling typically requires less RF power for a given bandwidth as compared to the conventional CPD sequences, this is often achieved at the expense of increased levels of decoupling sidebands. This can be a problem in spectra with high dynamic range such as NOESY where small peaks of interest may be obscured by decoupling sidebands from strong resonances. The bi-level CPD programs are designed to effectively suppress the decoupling sidebands and are relatively simple to set up. For example:

**cpdprg2:** bi\_p5m4sp\_4sp.2

userA1 sp14 (p62) :wvm:CHblv:f2 cawurst(150 ppm, 0.7ms; Q=4, L2H)

userA2 sp31 (p63) :wvm:CHdec:f2 cawurst(150 ppm, 1.4 ms; Q=3, L2H)

The effect of the bi-level decoupling is shown in [Figure 3.1](#) [▶ 12].

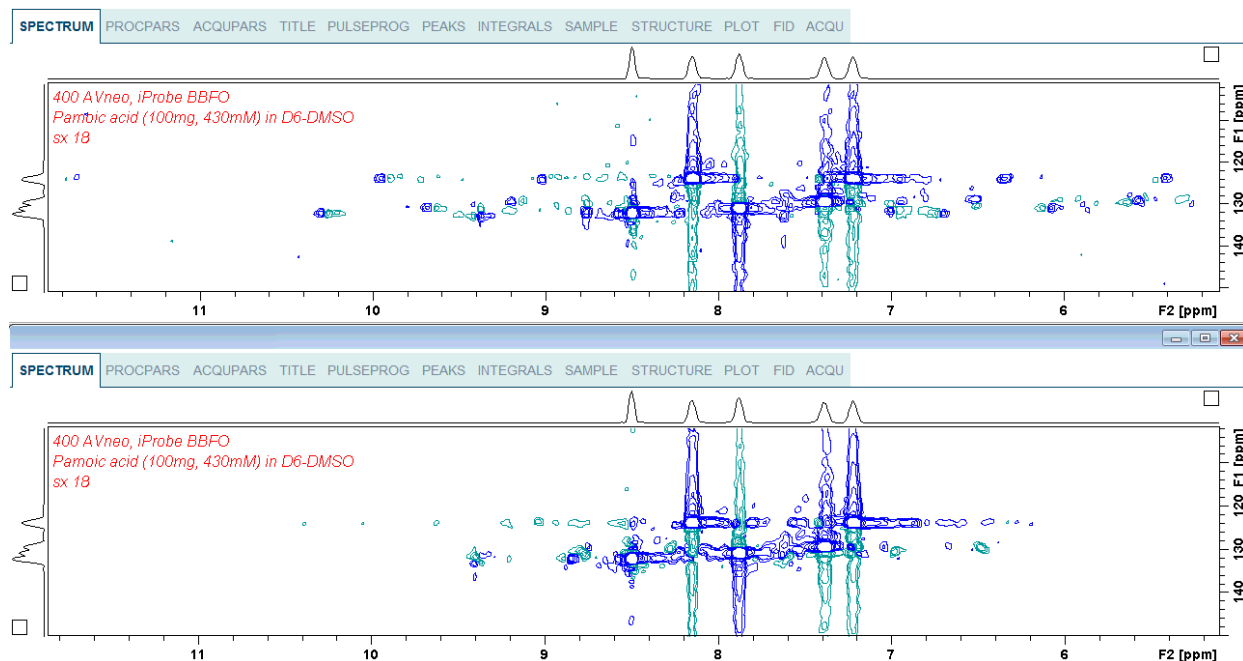


Figure 3.1: Decoupling sidebands in  $^{13}\text{C}$  HSQC spectra of pamoic acid in DMSO recorded using the standard pulse program, *hsqcetedetgpsp.3* and CPD program *p5m4sp180.p31* (top panel) and bi-level decoupling sequence *bi\_p5m4sp\_4sp.2* (bottom panel) with the decoupling sidebands effectively suppressed in just two scans per  $t_1$  increment.

## 3.5 Gradient Waveforms

The gradient waveforms are created in a very similar fashion to the RF waveforms, for instance:

```
gshape: sinc_g(5 G/cm, 1 ms) np = 250
gshape2: sinc_g() np = 100
gshape3: wurst_g(-80 %, 1 ms)
```

Note that in the first example the gradient strength is specified and the internally defined maximum gradient strength of ( $G_{\text{max}} = 53.5$  G/cm) is used for calibration purposes. In this case the gradient amplitude is set to achieve the same gradient pulse area as for 5 G/cm, 1 ms rectangular pulse. This allows using gradients independently of the specified shape function. If needed, the  $G_{\text{max}}$  can be set more accurately via the *wavemaker/wvm\_globals* file. The gradient waveforms are appended with the *.gp* extension and when created from TopSpin are stored in the *topspin/exp/stan/nmr/lists/gp/user* directory.

### 3.6 Adiabatic Pulses

Adiabatic pulses (see figure below) typically are frequency swept waveforms that unlike the conventional pulses allow simultaneous definition of both the bandwidth and the pulse duration as the shape function arguments. The arguments are recognized by the WaveMaker based on the units and their position. Note that the excitation bandwidth must always be defined first, i.e. before the (optional) pulse duration, offset and the phase which is always defined last. Occasionally the adiabaticity factor, Q or sweep direction, L2H - low to high field (positive to negative frequencies) or H2L - high to low field (negative to positive frequencies) may need be redefined as shown in the examples below:

```
cawurst-20(60 kHz, 1 ms)
cawurst-20(200 ppm, 1 ms; Q=4)
cawurst-20(200 ppm, 1 ms; L2H, Q=4)
```

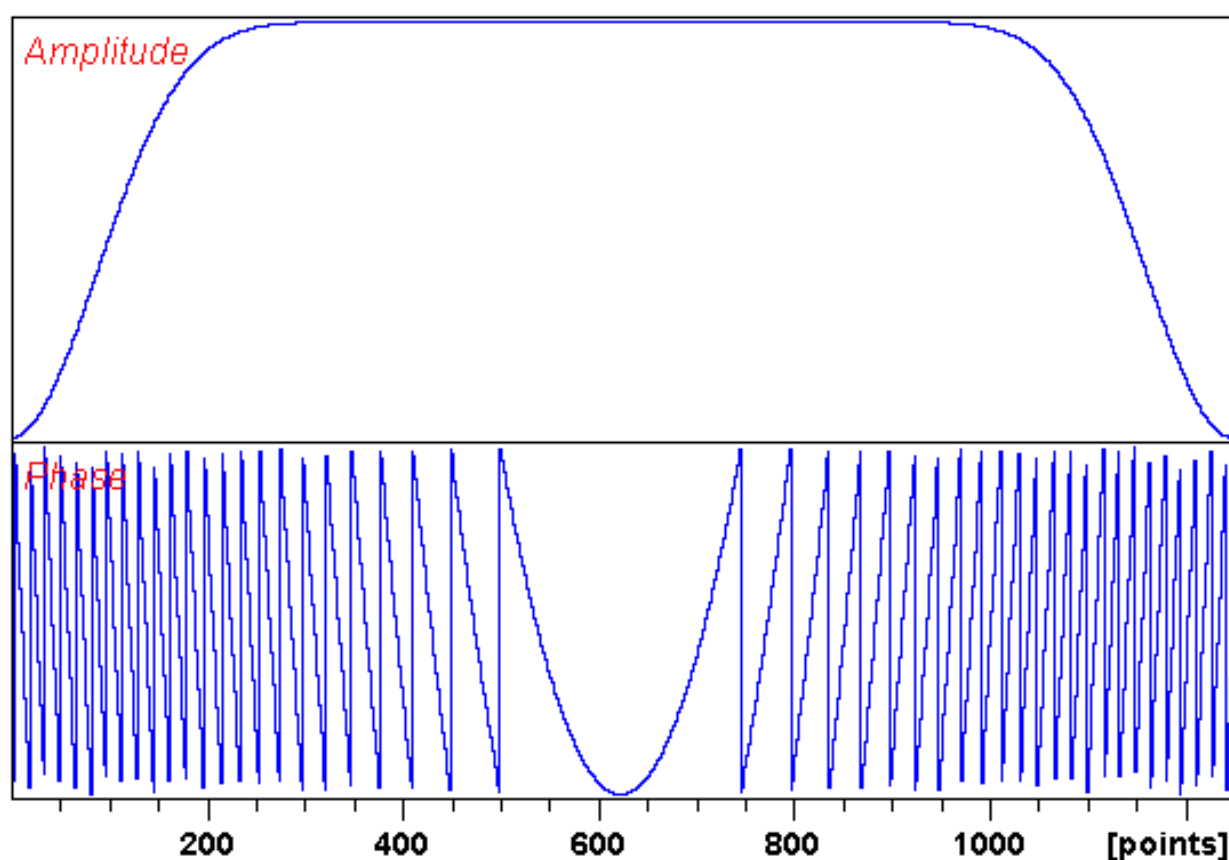


Figure 3.2: An example of adiabatic WURST-40 pulse with simultaneous modulation of both, the shape amplitude and frequency (phase).

### 3.7 J-Compensated Adiabatic Pulses

The J-Compensated adiabatic pulses are often used in C-13 HSQC based experiments, for instance for accurate J-refocussing and J-filtering (see C. Zwanen et al, *J. Am. Chem. Soc.*, **119**, 611, 1997; and Ě. Kupče and R. Freeman, *J. Magn. Reson.*, **127**, 36, 1997) and also for improved <sup>13</sup>C decoupling efficiency and sideband reduction. Such applications rely on the fact that there is an approximate correlation between the <sup>1</sup>J(CH) couplings and the C-13 chemical shifts, δ<sub>C</sub>. In proteins the following correlation has been found:

$${}^1J_{\text{CH}} = 0.365\delta_{\text{C}} + 120.0$$

The optimum pulse duration,  $T_p$  is calculated from the following equation:

$$T_p = 0.25 \cdot ({}^1J_{\max} - {}^1J_{\min}) / ({}^1J_{\max} \cdot {}^1J_{\min})$$

A linear sweep should be used for J-compensated pulses, for instance

```
wurst-40(280 ppm, 1.3 ms, 80 ppm) ss=1u
```

In order to obtain exactly 1.3 ms long pulse the step size, *ss* is set to 1  $\mu$ s. Since the bandwidth and offset are given in ppm, the above shape function is independent of the spectrometer frequency. Note that in a  ${}^1J_{CH}$  evolution period such pulses should be used immediately after the hard  ${}^1H$  180° pulse because by default the sweep direction is from high to low field, H2L (Alk to Ar). Alternatively, if the adiabatic  ${}^{13}C$  pulse is applied just before the hard  ${}^1H$  180° refocusing pulse the sweep direction can be set to L2H. For pulse programs where the adiabatic pulses are centered with the H-1 refocusing pulse the calculation is slightly different (see C. Zwaalen et al, *J. Am. Chem. Soc.*, **119**, 611, 1997).

The correct sweep rate and duration is calculated automatically by providing the Jcomp parameter, for instance:

```
wurst-20(250 ppm; Jcomp)
```

Since the default sweep direction is from Alk to Ar (H2L) this setting is correct for J-compensated adiabatic pulses applied immediately after the proton 180° pulse.

The correct J-coupling must be used in the experiment to ensure the best performance. The corresponding constant will be set into the parameters automatically with the *-a* option provided the following syntax is used:

```
"d2 = 0.25/cnst2" ;JCOMP
```

The default setting for the Jcomp parameter is Jcomp=1, which uses an internal setting for the A and B coefficients appropriate for peptides and proteins. For RNA / DNA samples the Jcomp parameter should be set to Jcomp=2. For instance:

```
wurst-20(250 ppm; Jcomp=2)
```

This will invoke an internal setting of the coefficients appropriate for RNA and DNA samples.

If the adiabatic pulse is applied immediately before the proton 180° pulse an opposite sweep direction is required. This can be specified as below:

```
wurst-20(250 ppm; Jcomp, L2H)
```

Users can specify their own J vs CS correlation coefficients using the C2 (A) and C3 (B) constants. For instance:

```
wurst-20(250 ppm; Jcomp, L2H, C2=0.302, C3=117.0)
```

Applications where the J-evolution delay is twice as long (e.g. J-filters) require setting C4 = 0.5. By default this coefficient is set to C4 = 0.25 as required for the INEPT type experiments.

For users who wish to make their own calculations, please note that the waveform definition is offset-dependent (see table below).

Offset, o1p (ppm)	Sweep bw (ppm)	Pulse duration, $T_p$ (ms)	$J_{\max}$ (Hz)	$J_{\min}$ (Hz)
70	200	0.92	182	109
100	220	0.877	197	116
110	260	1.014	208	113

100	160	0.618	186	127
100	200	0.788	193	120
100	240	0.97	200	113
80	160	0.682	178	120
80	200	0.872	186	113
80	240	1.077	193	105
80	280	1.3	200	98

Table 3.1:  $B_0$  Field-independent experimental parameters for  $J_{CH}$ -compensated pulses. The  $J_{max}$  and  $J_{min}$  are informative and not used experimentally.

While the parameters in the table above are  $B_0$  field independent the actual inversion bandwidth will depend on the absolute sweep bandwidth,  $bw$  (in kHz) and is typically significantly smaller. The profiles become less 'rounded' at higher  $bw \cdot T_p$  product values (higher  $B_0$  fields). The inversion profile of a J(CH) compensated WURST-40 pulse on an 800 MHz NMR system is shown in the figure below.



**For adiabatic  $^{13}\text{C}$  decoupling the optimum sweep direction is L2H**

(from Ar to Alk)!

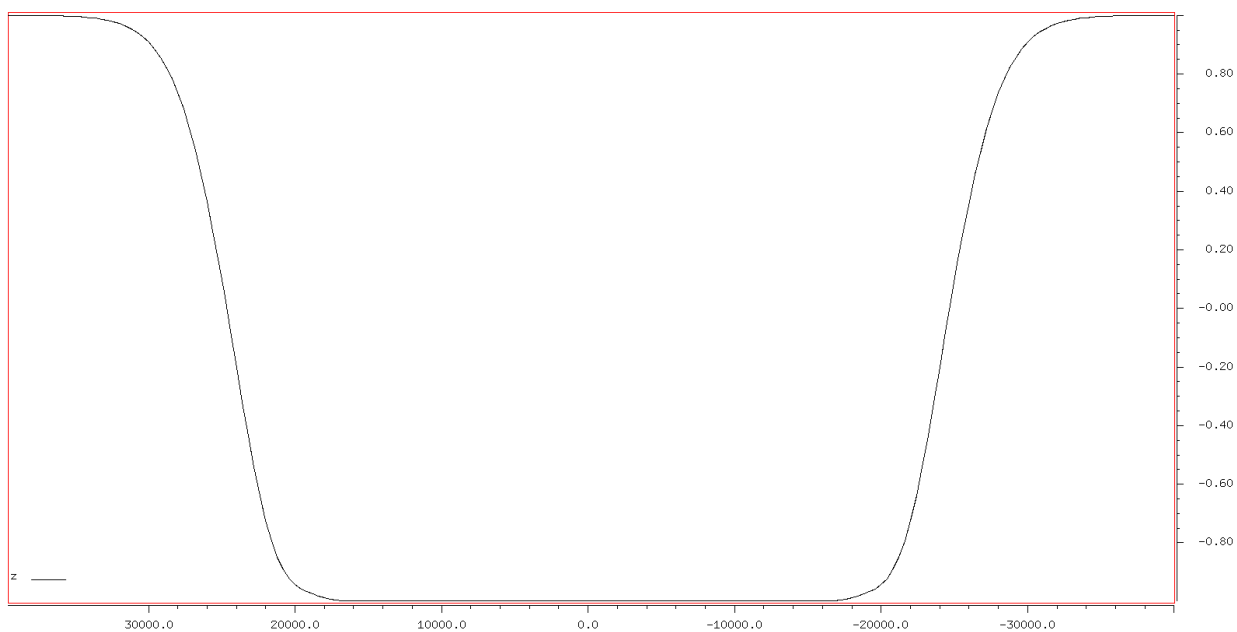


Figure 3.3: Inversion profile of an adiabatic WURST-40 pulse with sweep bandwidth set to 56 kHz and pulse duration to 1.3 ms. Note that the effective inversion bandwidth is only ca 44 kHz.

### 3.8 Composite Adiabatic Pulses

The composite (1:2:1) adiabatic pulses can be generated by specifying the [comp121] phase cycle in the phase argument space (the last argument in the shape function). In order to distinguish it from a phase encoding scheme it is recommended to use the square brackets, for instance:

```
cawurst-20(80 kHz, 2 ms, [comp121])
```



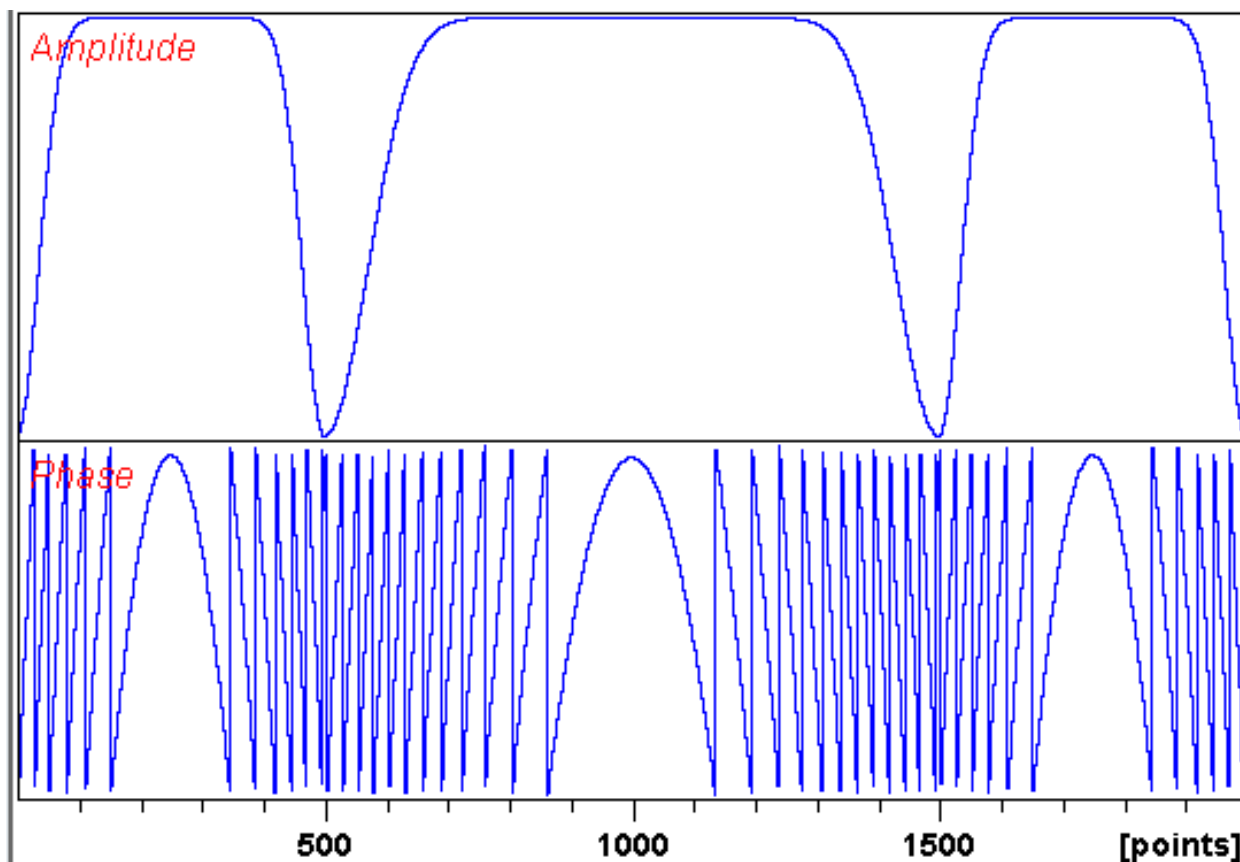


Figure 3.4: An example of a composite adiabatic pulse consisting of three cawurst-20 pulses of total duration set to 2 ms and the sweep bandwidth to 80 kHz as describe in the main text.

### 3.9 Adiabatic pulses in $^{19}\text{F}$ NMR

There are many applications in  $^{19}\text{F}$  NMR that require use of wideband RF pulses. The CPMG experiments that are of particular interest in drug screening experiments require wideband refocusing pulses that must cover bandwidths of 100 ppm or more ([Figure 3.5 \[p 17\]](#)). Such experiments often need to be recorded in automation. Adiabatic pulses provide a convenient solution to these demanding experiments.



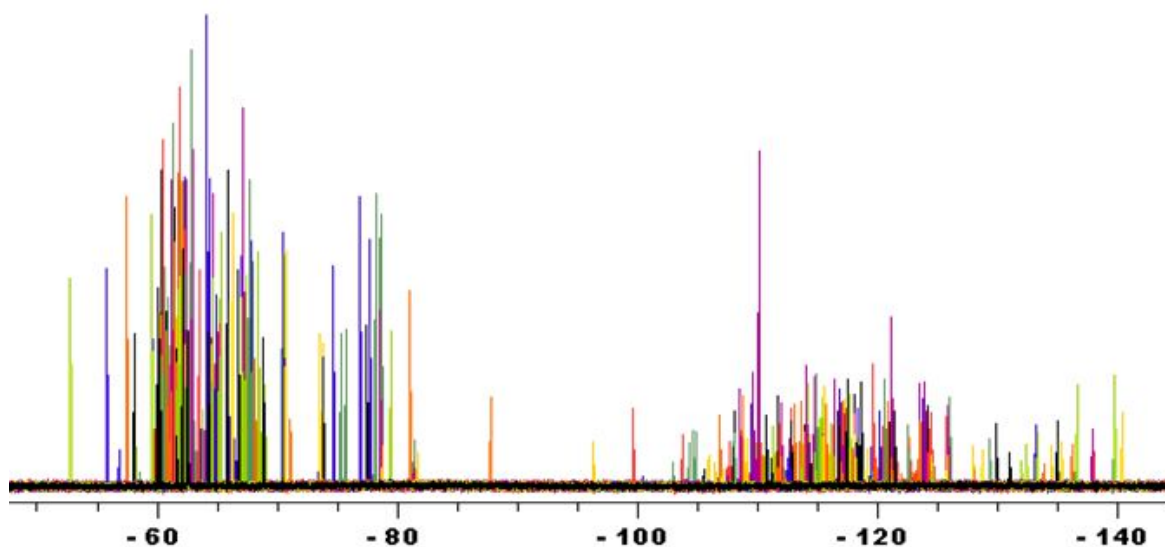


Figure 3.5:  $^{19}\text{F}$ -spectra of 24 cocktails of 460 compound library from Keyorganics. The ca. 100ppm bandwidth needed for  $^{19}\text{F}$ -experiments corresponds to 66kHz at 700MHz. Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland).

Such pulses are conveniently generated using the following wavemaker function:

```
;sp5:wvm:cawurst-20(cnst1 ppm, 2 ms)
```

where `cnst1` parameter represents the sweep bandwidth for the adiabatic pulse. By setting `cnst1` to 140 ppm the desired bandwidth of 92 kHz ensures > 99% inversion efficiency over the active bandwidth (see Figure 3.6). In automation experiments the required shaped pulse is generated by including the following line in the AUNM au-program:

```
XCMD('wvm -q')
```

In routine applications the probe-dependent parameters first are set using the **getprosol** command. The required shapes can then be generated using the **wvm -a** command. The created shapes can be inspected using the **wvm -b** command ([Figure 3.6](#) [▶ 18j](#)).

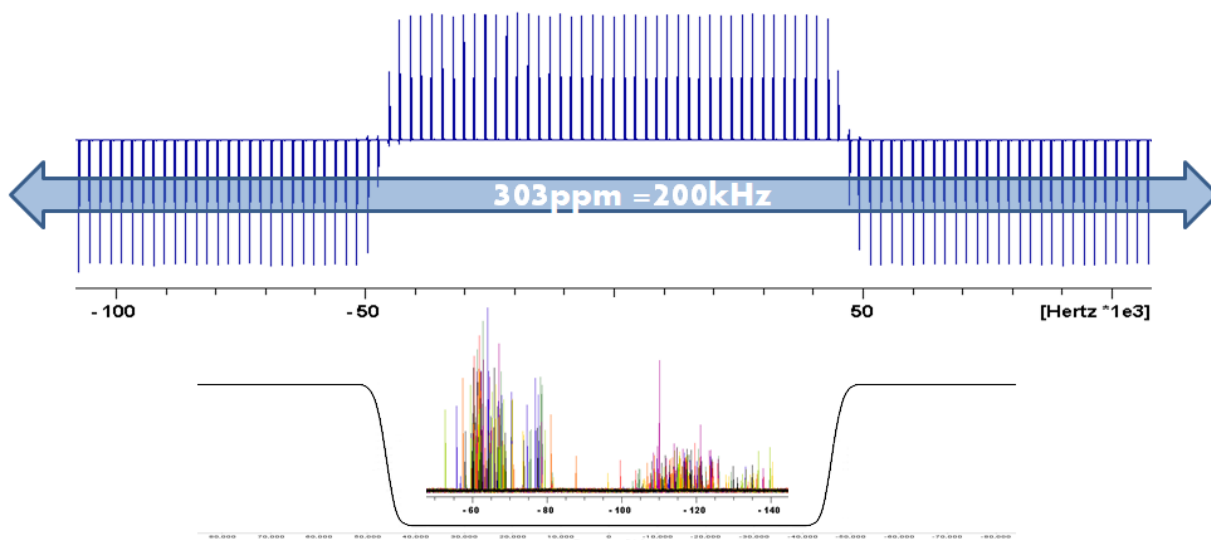


Figure 3.6: The experimental and simulated profiles of the adiabatic pulse generated using the shape cawurst-20 (140 ppm, 2 ms) on a 700 MHz instrument. The active bandwidth covers 92 kHz with > 99 % efficiency using 6.95 kHz RF peak amplitude (corresponding to 36  $\mu$ s  $90^\circ$  square pulse). Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland).

Adiabatic pulses are extremely robust to RF inhomogeneity and mis-calibration.

### 3.10 Notch Pulses

The notch pulses are simple pulses with a null at a given distance from the center of the main band. These pulses are named by adding a letter 'n' at the end of the name, for instance square90n or square180n. The distance between the center of the main excitation band and the position of null is given in the bandwidth field of the shape function, e.g. square90n (1000 Hz) (see the Mxy excitation profile below).

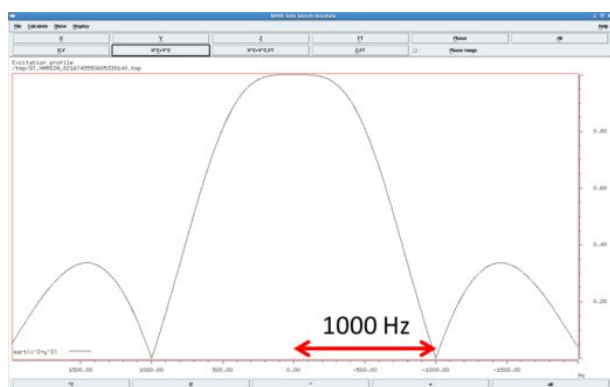


Figure 3.7: The excitation profile of the square90n notch pulse showing the special bandwidth definition that indicates the position of the first zero crossing.

### 3.11 Optimized Notch Pulses (NEW)

The position of the zero crossing in notch pulses may slightly vary depending on the flip angle (see figure below). In many applications these minor changes are of little concern because typically a fixed on-resonance rotation angle of  $90^\circ$  or  $180^\circ$  is used and the shapes are well calibrated for that particular flip



angle. However, in some applications (such as SOFAST and BEST techniques or similar) notch pulses with arbitrary flip angles may be required. For instance, it may be useful to set the notch position at the solvent frequency in order to achieve better solvent (e.g. water) suppression. In such circumstances even a slight variation of the notch position can significantly degrade the quality of the data. Therefore the notch pulses are automatically calibrated to achieve the exact notch position at any given flip angle. The parameter (VN) that activates the internal calibration is set in the shape definition files in the wavemaker/shapes/notch directory.

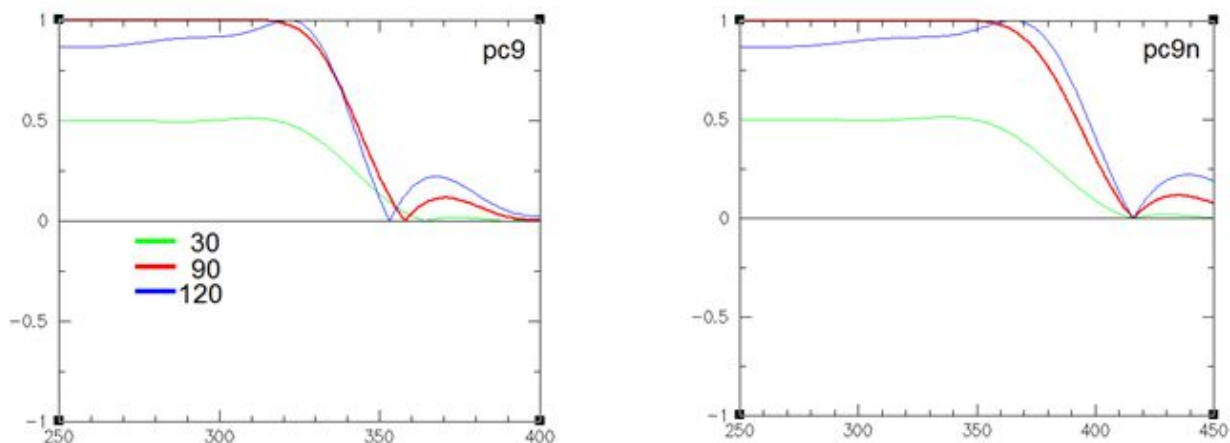


Figure 3.8: The position of the zero crossing is slightly changing in the pc9 pulses (left panel) depending on the flipangle set to 30, 90 and 120 degrees. In contrast, it is independent of the flip angle in the notch pulse pc9n (right panel).

### 3.12 Time Shifted Pulses (NEW)

When several shaped pulses are combined their individual points of origin can be shifted in time. This is achieved using the delay, *d1* that defines the position of that particular shaped pulse with respect to the beginning of the whole waveform. An example is shown below.

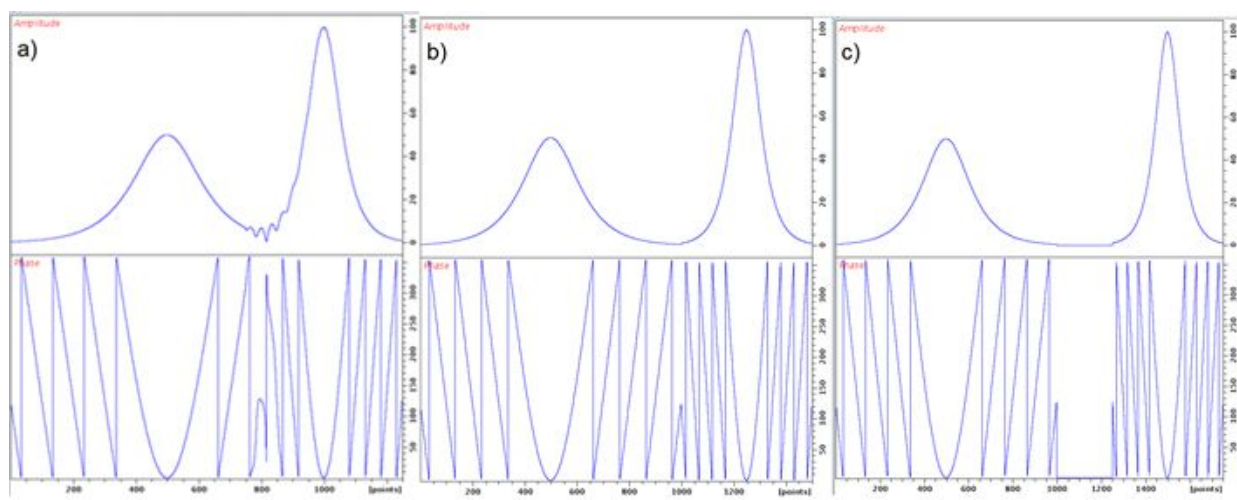


Figure 3.9: Three examples of shifting the origin of one of the shaped pulses in a waveform combining two hyperbolic secant pulses



The following syntax is used for creating the waveforms shown in [the figure above](#) [ 19]:

```
sech(10 kHz, 1 ms; d1 = 1.5 ms) sech(10 kHz, 2 ms)
sech(10 kHz, 1 ms; d1 = 2.0 ms) sech(10 kHz, 2 ms)
sech(10 kHz, 1 ms; d1 = 2.5 ms) sech(10 kHz, 2ms)
```

### 3.13 Shape Definition Files

The shape definition files are text (ascii) files that are stored in the wavemaker/shapes directory. The shapes are grouped in sub-directories according to their applications: **excitation, inversion, refocusing, flipback, decoupling, spinlocks, phasecycles** and **gradients**. The shape parameters can be adjusted either (a) by modifying the shape file contents in the shape directory (bad practice) or (b) by submitting the modified parameters via the shape functions (good practice). The modified parameters are separated from the shape arguments by a semicolon, for instance:

```
cawurst-20(200p, 1.2m; Q=4, H2L)
eburp2(50 Hz; np=1000)
```

Note that unless the modified shape definition files are also provided (case a) the changes remain hidden and unavailable to other users who may be using the provided WaveMaker input files. This may produce inconsistent results and the practice is generally discouraged.

On the other hand, the WaveMaker shape library can be extended with new shape definition files when such shapes become available either from literature or designed by the users. The shape definition parameters are shown in [Appendix](#) [ 38].

An easy access to the shape directory contents is provided via the `-i` option, for instance: `wvm -i eburp1`

### 3.14 Using TopSpin Parameters

The TopSpin `cnst#` parameters can be used for convenience and to avoid explicit definitions of WaveMaker variables in the pulse sequences:

```
sinc90(cnst22 ms)
wurst2(cnst21 kHz, cnst25 ms; Q=3.0)
eburp1(cnst15 ppm)
ofs = cnst17 ppm
```

Note, however, that constants will not be displayed in the pulse program parameter display unless they are used in calculations within the pulse program. One possible workaround is to use a dummy calculation, e.g.

```
"cnst17 = cnst21 + cnst22 + cnst15"
```

Up to five constants can be used on the right hand side of such equation.

In some applications, such as spin locks or water flip-back pulses, a preset pulse duration is required. In such cases the pulse name can be given as an argument to the shape function, for instance:

```
tanAM(p34; Blav=cnst32 kHz)
```

### 3.15 Re-defining the Phase Alignment (PA)

The PA parameter in the WaveMaker is aligned with the TopSpin pulse shaping PA parameter, meaning that for phase ramped (off-resonance) pulses PA=0 ensures that the phase is aligned at the beginning of the waveform, which is what is required for proper functioning of the flip-back pulses. For refocusing pulses the phase alignment is set to PA=0.5 and for the excitation pulses to PA=1.0. For inversion (and decoupling) pulses PA is usually of no importance, since the magnetization is rotated from  $M_z$  to  $-M_z$  or vice versa. In WaveMaker the PA is set automatically depending in which directory the shape resides, excitation (PA=1.0), refocusing (PA=0.5) or flip back (PA=0.0). In addition, PA is automatically set to PA=0 for time-reversed pulses (TR=1) and to PA=0.5 for adiabatic pulses in order to make sure they function properly. In all other cases PA is defaulted to PA=1.0. The user can always redefine this parameter as necessary, for example:

```
G3(25 ppm, 56 ppm; PA=0.5)
```

In any case, note that PA is only active for phase ramped (off-resonance) pulses and when shapes of different duration/bandwidth are combined, i.e. applied simultaneously.

### 3.16 Re-defining the Excitation Bandwidth

All shapes have internally defined excitation bandwidth (see BW parameter in shape definition files). This usually defines the excitation, inversion or refocussing bandwidth at 80% level in accordance with majority of definitions found in literature. Unfortunately, such definitions are far from unanimous. Furthermore, for pulses with variable active bandwidth (e.g. adiabatic pulses) the BW used to define the frequency sweep range, which for constant adiabaticity pulses corresponds approximately to the active bandwidth at 50 % level. For other forms of adiabatic sweep this parameter is even less related to the active bandwidth. In order to allow accurate setting of the active bandwidth the *bw/vl* parameter can be used, for instance:

```
cawurst-8(8 kHz, 1.5 ms; bw/vl = 0.8Mz).  
eburp1(4 kHz; bw/vl = 0.9Mx).  
pc9(4.2 ppm, 8 ppm; bw/vl = 0.9Mxy)
```



Note, that since the shape specific parameters typically are unitless, there is no space between the numerical setting and the profile name and this syntax must be strictly observed (i.e. 0.9Mz and not 0.9 Mz).

### 3.17 Multi-frequency Waveforms

Simultaneous multiple excitation waveforms can be generated by simply defining several shape functions, for instance symmetric inversion pulses with respect to the 58 ppm offset (C-alpha) can be applied using the following shape definition:

```
;sp17:wvm: iburp1(30 ppm, 176 ppm) iburp1(30 ppm, -55 ppm)
```

Alternatively two excitation bands that are symmetric with respect to the carrier frequency can be implemented using amplitude modulation flag, AM set to AM=cos or AM=sin:

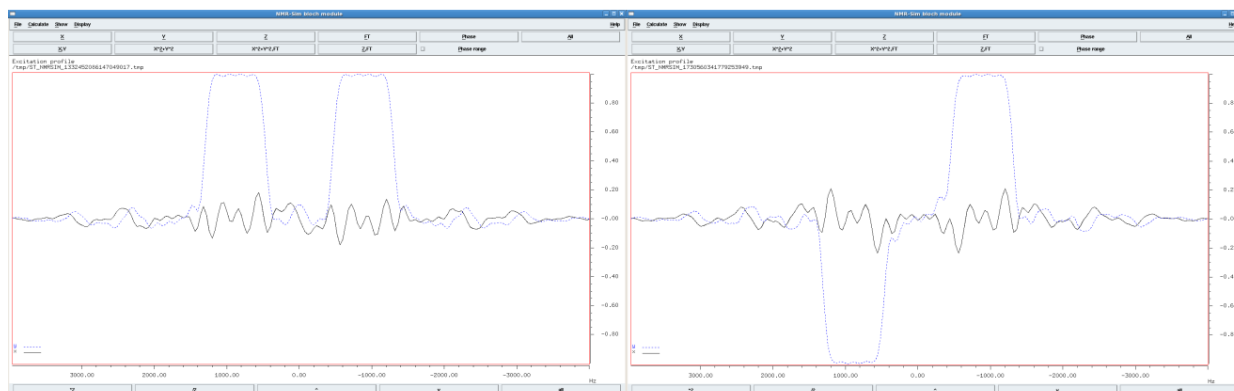


Figure 3.10: Bloch simulations for double excitation waveforms created using the AM flag and the following shape functions - the left panel: `eburp1(0.8 ppm, 3.715 ppm; AM=cos)` and the right panel: `eburp1(0.8 ppm, 3.715 ppm; AM=sin)`; In this example `o1p = 4.7 ppm` and `BF1 = 900 MHz`.

### 3.18 Shape Arrays

Several waveforms can be combined into a single shaped pulse simply by specifying several shape functions for simultaneous excitation, e.g.

```
reburp(25 ppm, 176 ppm) reburp(25 ppm, 56 ppm)
```

Alternatively a peak list can be used, e.g. for multiple peak (solvent) suppression. The following workflow may be used to achieve that.

- Create a peak list in experiment 55.
- Create a new experiment, e.g. 56. In the pulse sequence (for instance `selzg`) add the following line to the comments section:  

```
;sp1:wvm:myshape: gaus90(const21 Hz, PL)
```

 Set `const21` to the required excitation bandwidth, e.g. `const21 = 50 Hz`;
- Copy the peak list from experiment 55, e.g. `getPL 55`  
 The `getPL` au-program is available from the online User Library.
- Execute the `wvm` command, e.g. `wvm -a`. Check the output, e.g. `wvm -l`. If everything is fine type `zg` to start the experiment.

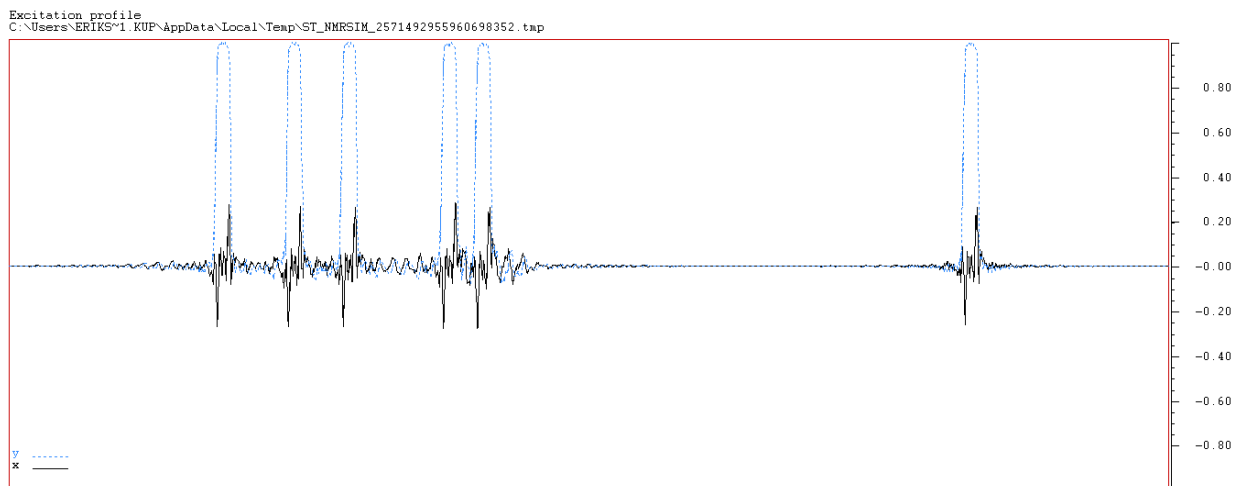


Figure 3.11: The excitation profile of the shape created as described above using E-BURP-1 pulses.

### 3.19 Compensation of Bloch-Siegert Interferences

The evolution of spin magnetization in the presence of RF irradiation (shaped pulses and decoupling waveforms) is influenced by the Bloch-Siegert effect. Consequently, when more than one shaped pulse is executed at the same time the excitation profiles are distorted to an extent that strongly depends on the distance between the offsets of the individual shaped pulses as well as their RF amplitudes. In all cases when more than one shaped pulse is executed simultaneously such distortions are automatically compensated for when the shapes are created by WaveMaker. The effect of BS compensation is shown below for the iburp2 pulses:

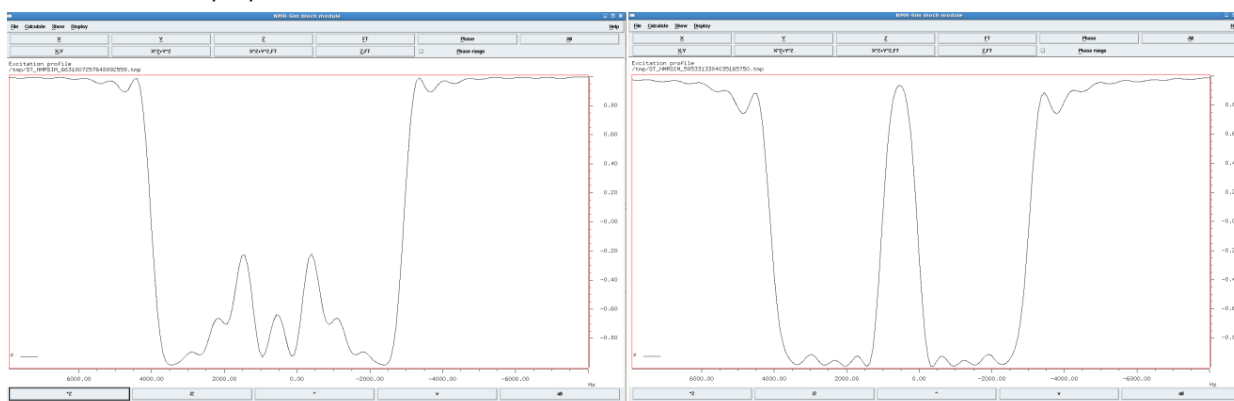


Figure 3.12: The effect of BS compensation on double excitation profiles of iburp-2 pulses; Left – with BS compensation disabled (iburp2(3 ppm, 7.6 ppm) iburp2(3 ppm, 3.0 ppm) BS=0) and right – with automatic BS compensation (iburp2(3 ppm, 7.6 ppm) iburp2(3 ppm, 3 ppm)). The offset was set to o1p = 4.7 ppm and BF1 was 900 MHz.

The interference between simultaneous asymmetric RF pulses can be further reduced by time-reversing one of the shapes. This can easily be achieved in the case of inversion pulses and in some cases also for excitation pulses, for example if one of the pulses is used for de-excitation. The additional effect of time reversal is shown below.



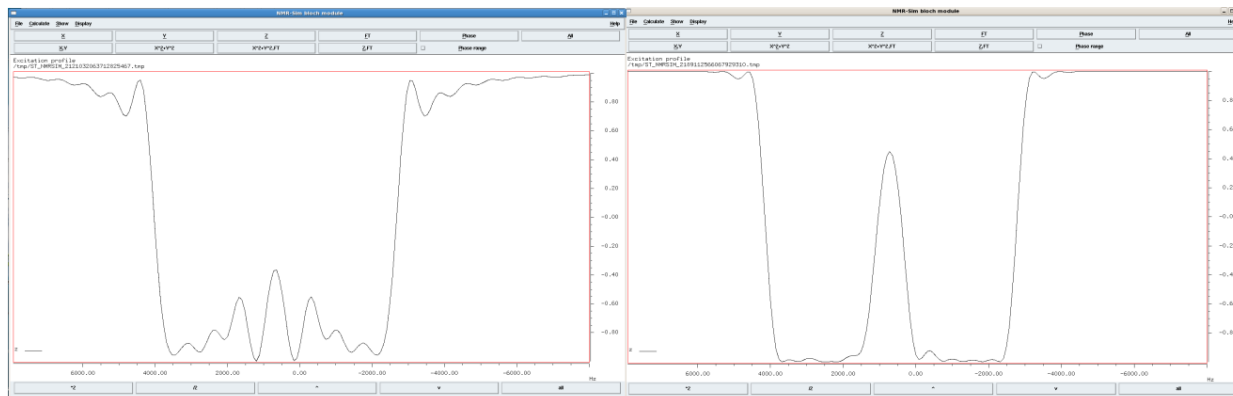


Figure 3.13: The effect of time reversal of one of the inversion pulses on the profile of double inversion with *iburp-2* pulses.

Note that the excitation bands are brought together even closer than in the previous example leaving only a small gap between the two inversion profiles. On the left, the BS compensation alone does not completely restore the required excitation profiles (*iburp2*(3 ppm, 7.6 ppm) *iburp2*(3 ppm, 3.3 ppm)). On the right one of the shapes is time-reversed and combined with automatic BS compensation restoring the required inversion profiles (*iburp2*(3 ppm, 7.6 ppm) *iburp2*(3 ppm, 3.3 ppm; TR)). The offset was set to 4.7 ppm and BF was 900 MHz.

If necessary (e.g. for comparison), the BS compensation can be disabled by resetting the BS flag to BS = 0. For instance:

```
;sp17:wvm: iburp1(30p, 176p) iburp1(30p, -55p) BS=0
```

### 3.20 Using Peak Lists and Hadamard Encoding Experiments

Experiments that require use of multiple shapes and incremented waveform files, e.g. multiple solvent suppression experiments, Hadamard experiments, phase-encoded experiments and similar, are conveniently generated using peak lists. All the topspin list formats are accepted and the list file names are provided in the corresponding field in the shape function. The encoding type is defined in the phase definition field. For example, the Hadamard encoding pulses are generated using the following definition:

```
eburp1(cnst21, PL, had)
```

Where the bandwidth parameter, *cnst21* defines the excitation bandwidth for the encoding pulses, typically 10-50 Hz, the offset parameter, *PL* is the name of the peak list, *PL.xml* and *had* indicates that Hadamard phase encoding is required. Note that the peak list file *PL.xml* must reside in the acquisition directory rather than in processing directory, *pdata* where it is typically created. It can be fetched from the experiment (typically 1D) where it was created using the *getPL* au-program, e.g. *getPL 11*, where *11* is the experiment number containing the original peak list. Furthermore, note that the extension of the peak list file, *PL* in the shape function in this case is omitted. To simplify the syntax *PL* is used as a keyword in WaveMaker that associates the *PL* entry as *PL.xml* filename. In all other cases the filename extension is expected.

In pulse programs (TopSpin 3.2 and later) the Hadamard encoding is implemented by defining shape lists, *SPL*:

```
define lists <shape> SPL=<${VALIST}>
```

The list is then executed as a shaped pulse, for instance:

```
(p34:SPL ph1):f1
```



The list index is then typically incremented as follows:

```
d11 mc #0 to 2 F1QF(SPL.inc)
```

Therefore, the complete WaveMaker syntax for generating shapes that are required for Hadamard experiments is as follows:

```
;SPL:wvm: eburp1(cnst21 Hz, PL, had)
```

The optional shape file name is omitted here for simplicity. When the **wvm** command is executed the following output is generated:

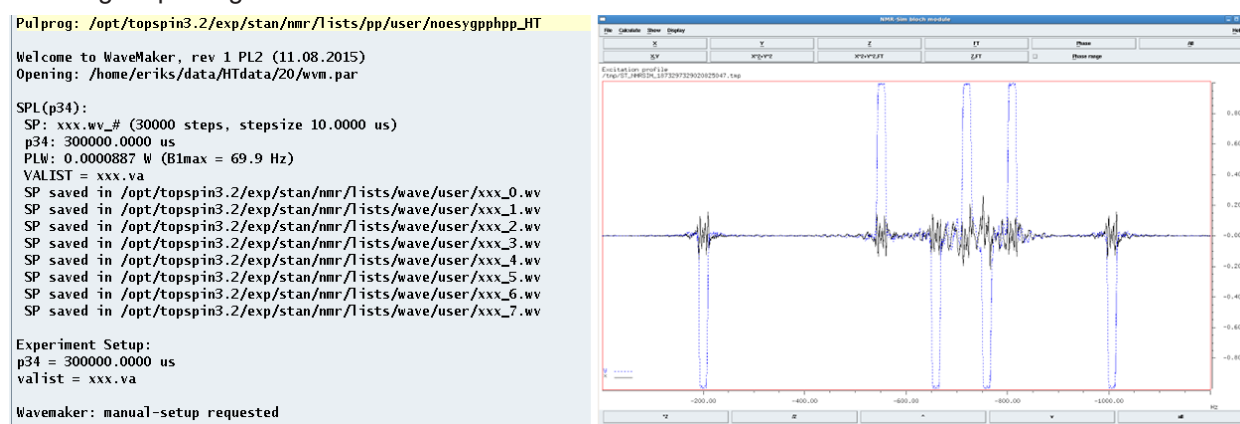


Figure 3.14: The left panel - WaveMaker output in a Hadamard encoding experiment with  $cnst21=15.0$  and the following shape definition: `SPL:wvm:xxx: eburp1(cnst21 Hz, PL, had)`; the right panel – Bloch simulation for the second encoding shape (`xxx_1.wv`).

Note that the VALIST file is automatically generated along with all the required shape files and stored in the lists/sp directory. Note (!) - the implementations differ in previous releases of TopSpin.

In general, WaveMaker allows for four types of encoding: simultaneous, *sim* (default), sequential, *seq*, Hadamard, *had* and phase encoding, *ph*. In the first case (encoding = *sim*, by default) a single waveform is created that combines multiple excitation patterns. In the second case (encoding = *seq*) N waveform files containing a single frequency shape per file are created where N is defined by the flist (blist, plist).

The Hadamard encoding (encoding = *had*) requires a frequency list that is used to generate M shaped files combining N waveforms according to the Hadamard phase encoding matrix as described above. In this case M is the nearest power of two that is larger than N, where N is defined by the flist.

Phase encoding (encoding = *ph*) is an alternative to Hadamard encoding with an advantage that the matrix size is the same as the number of the encoded frequencies. The disadvantage is that both the real and imaginary parts of the signal need be recorded and the sensitivity advantage is generally lost. However, it can be recovered in so called sensitivity enhanced experiments that record both the real and imaginary parts of the signal simultaneously. Nevertheless, 2N experiments still need be recorded. The waveforms needed for phase encoded experiments can conveniently be generated using the WaveMaker with the encoding set to 'ph', for instance:

```
eburp1(cnst21, PL, ph)
```

WaveMaker then generates N waveforms combining N shapes according to the phase encoding matrix where phases are encoded as  $n*i*360/N$  (experiment number,  $i = 0, 1, \dots N-1$ ; frequency number  $n$  is fixed for each entry in the flist,  $n = 0, 1, \dots N-1$ ).

## User-defined encoding matrices

Wavemaker accepts user provided encoding matrices stored in the current experiment as a text file **wvm.had** in the following format:

```
H12
+-----
++-+---++-+
+++-+---+++
+-++-+---+++
++-++-+---++
+++-++-+---+
++++-++-+---
+-++++-++-+-
+---++++-++-+
+----++++-+++
++++-++++-+-
+-+----++++-++
```

Here H12 is the encoding matrix size. The same file is then used by **proc\_had** au-program for processing. Note that the matrix rows are used for encoding while columns are used for re-construction (Hadamard transform).

## Two-dimensional peak lists

The two-dimensional peak lists are handled in the same way as the 1D peak lists. The 2D peak lists are imported and saved as **PL.xml** file with the updated **had\_pl** au-program which also sets the TD(1) parameter according the peak list size. The WaveMaker generates **ht.pkl** file that is used by **proc\_had** processing routine. In experiments that use Hadamard encoding in two dimensions a second file, **ht1.pkl** is generated that can be used for 3D reconstruction.

## Compensation for temperature fluctuations - B1comp

Hadamard encoding experiments involving large number of on/off encoding pulses (inversion or saturation) may experience slight temperature fluctuations because of the varying number of active RF components in the Hadamard encoding matrix. To compensate for such variations in power dissipation, off-resonance RF pulses are applied when the Hadamard encoding pulse (RF component) is in the “off” mode. The RF compensation is activated by the B1comp flag, for instance:

```
cw_i(800 ms, PL, had; B1max = 6 Hz, B1comp = 10 kHz) ss = 5 us
```

In this example 800 ms long cw saturation is used in on/off mode with the B1 field of 6 Hz and step size of 5 us. Two compensating RF fields of half the encoding pulse amplitude are applied at  $\pm 10$  kHz off-resonance from the carrier frequency to keep the total RF power dissipation constant and to minimize the effects of Bloch-Seigert shifts caused by single frequency off-resonance irradiation.

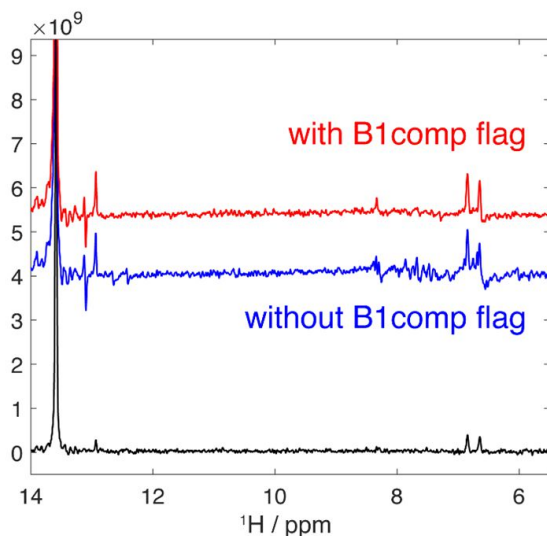


Figure 3.15: The effect of B1comp flag on artefacts in Hadamard encoding experiment. Spectra are courtesy of Dr. Jihyun Kim and Prof. Lucio Frydman (Weizmann Institute).

### 3.21 Multiple (Solvent) Peak Suppression

Multiple solvent or other unwanted peak suppression is easily achieved using peak lists as discussed in sections 3.18 [▶ 22] and 3.20 [▶ 24], except in this case multiple frequency pre-saturation waveforms are employed. The syntax for such shapes involves the `cw()` shape function. For instance:

```
;sp6:wvm:mfpresat: cw(1s, PL; B1max = cnst6 Hz)
```

The constant `cnst6` in this example is used to control the pre-saturation RF field strength. The `PL.xml` file containing the list of peaks to be suppressed must reside in the current experiment and can be copied using the `copy_pl` au-program from an experiment where the peaklist has been previously generated. Note that all frequencies in the frequency list will experience the same RF field strength.

Occasionally the users may want to apply a stronger RF field to a particularly strong peak, such as the water resonance which may be more difficult to suppress, for instance due to the radiation damping effects or a broader linewidth. This is achieved by adding an additional RF waveform applied to that frequency with a different RF field strength:

```
;sp6:wvm: cw(1s, PL; B1max = cnst6 Hz) cw(1s; B1max = cnst7 Hz) ss = 2 us
```

In this example the two waveforms are combined and the RF field strength applied to the on-resonance water frequency is optimized individually using the `cnst7` parameter. Note also that the missing file name means that the shapefile name will be generated automatically. The optional stepsize parameter (`ss = 2 μs`) may be specified to avoid excessive number of points being generated on older systems with limited waveform memory. An example of multiple peak suppression using two simultaneous waveforms is shown in Figure 3.16 [▶ 28].

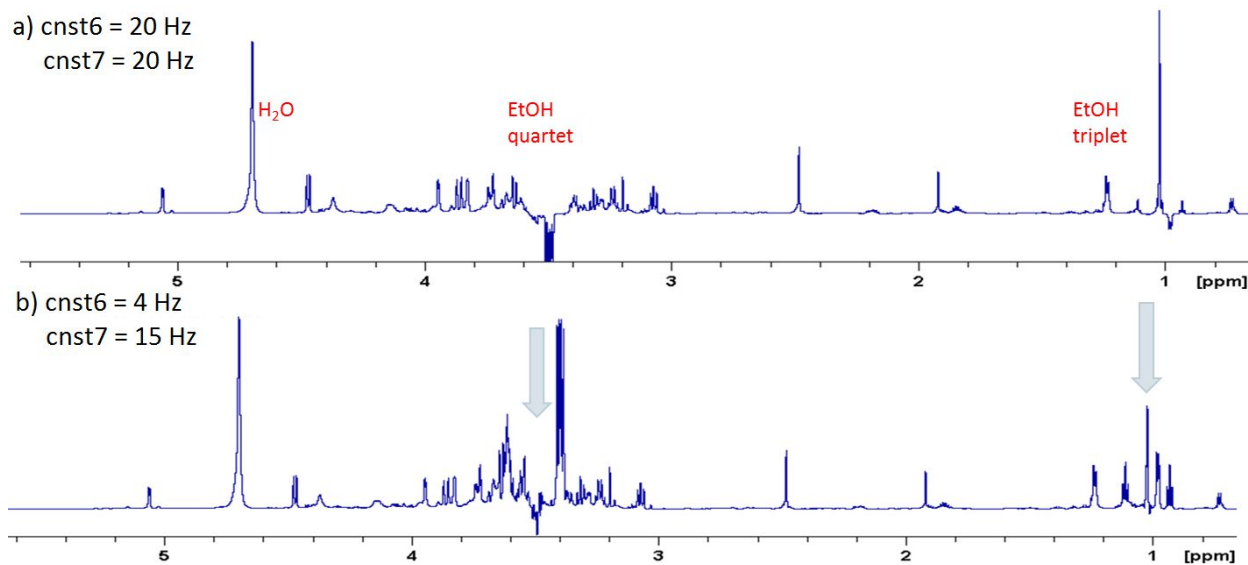


Figure 3.16: Multi-frequency peak suppression using two `cw()` waveforms as described in section 3.21 [▶ 24]. The RF field strength applied to the EtOH peaks is defined by `cnst6` and to the on-resonance H<sub>2</sub>O peak by `cnst7`. Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland).

### 3.22 Adiabatic Homo-decoupling

Adiabatic homo-decoupling is often used in protein experiments, for instance, to decouple C-alpha-s from C-beta-s centered at 30 ppm (30 ppm bandwidth) and 70 ppm (4 ppm bandwidth). Note that currently WaveMaker does not check for any frequency jumps that may occur in the pulse program. Therefore, the user must account for such jumps and define the correct frequency offset (if different from `o2p` or similar) as shown below:

```
;cpds2:wvm: cawurst_d-8(30 ppm, 5 ms, 30 ppm; H2L, Q=1.2, sucyc=p5)
;wvm: cawurst_d-2( 4 ppm, 5 ms, 70 ppm; L2H, Q=1.2, sucyc=p5)
;wvm: ofs=cnst21 ppm, ss=5.0 us;
```

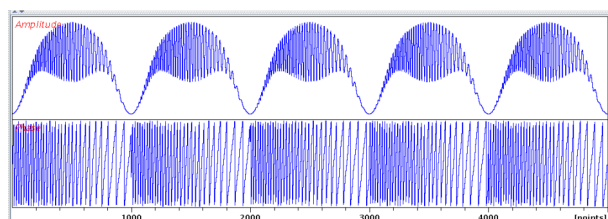


Figure 3.17: The homo-decoupling waveform generated by WaveMaker as described in the main text above.

Indicating the offset units (ppm in this case) is important since the default unit is Hz. The `cawurst_d` (constant adiabaticity WURST) shape definition file is found in `shapes/decoupling` and the amplitude power index for the WURST shapes (i.e. 8 and 2) is indicated via dash. In this example the power index of 8 for the main beta-band is used to reduce the modulation sidebands, whereas for the smaller band at higher frequencies the power index of 2 is used because it provides for a better inversion profile in the case of narrow frequency range. The relatively low adiabaticity factor ( $Q=1.2$ ) and the opposite sweep direction for the two bands helps reducing the modulation sidebands. Specifying the step size (`ss`) is optional.



### 3.23 Specifying the Required B1max, B1rms and B1av

Instead of specifying the desired bandwidth or pulse duration of shaped pulses, decoupling waveforms or spin locks, the users can specify the desired B1max (B1rms or B1av) as a shape parameter. For instance,

```
eburp1(200 Hz, -1.5 kHz; B1max= 1.0 kHz)
```

In this case the required pulse duration is calculated and the bandwidth is defined depending on the fixed pulsewidth to bandwidth product. In the case of adiabatic pulses, the user has a choice of keeping the pulse duration or the bandwidth unaffected. For example:

```
sech(1.0 ms; B1max= 10.0 kHz)  
sech(20 kHz; B1max= 10.0 kHz)  
sech(20 kHz, 1.0 ms; B1max= 10.0 kHz)
```

In the first case the pulse duration is kept constant and the sweep bandwidth is adjusted to match the required RF amplitude. In the second example the bandwidth is kept constant, while in the third example both pulse duration and bandwidth are adjusted. In the case of some spin lock waveforms the rotation angle is undefined and the RF amplitude is simply set to the required magnitude. Similar rules apply to setting of the user defined RMS (B1rms) and the average (B1av) RF amplitude.

### 3.24 Off-resonance Spin Locks

Simple off-resonance spin locks are amplitude modulated RF ramps that can be applied either 'on-resonance' (no phase ramps) by jumping the frequency or 'off-resonance' by using the phase ramps. Typically the steepness of the initial and final ramps needs adjustment in order to satisfy the adiabaticity requirements. A few simple spin locks are shown below.

- wurstAM – WURST-shaped spin lock; the steepness of the initial and final stages of the spin lock is controlled by the power index, e.g. wurstAM-80 (the power index is set to 80). In this example the spin lock is implemented 'on-resonance' by a frequency jump (no phase modulation):

```
;sp14:wvm:SL1: wurstAM-80(100 ms; B1max= 5.2 kHz)
```

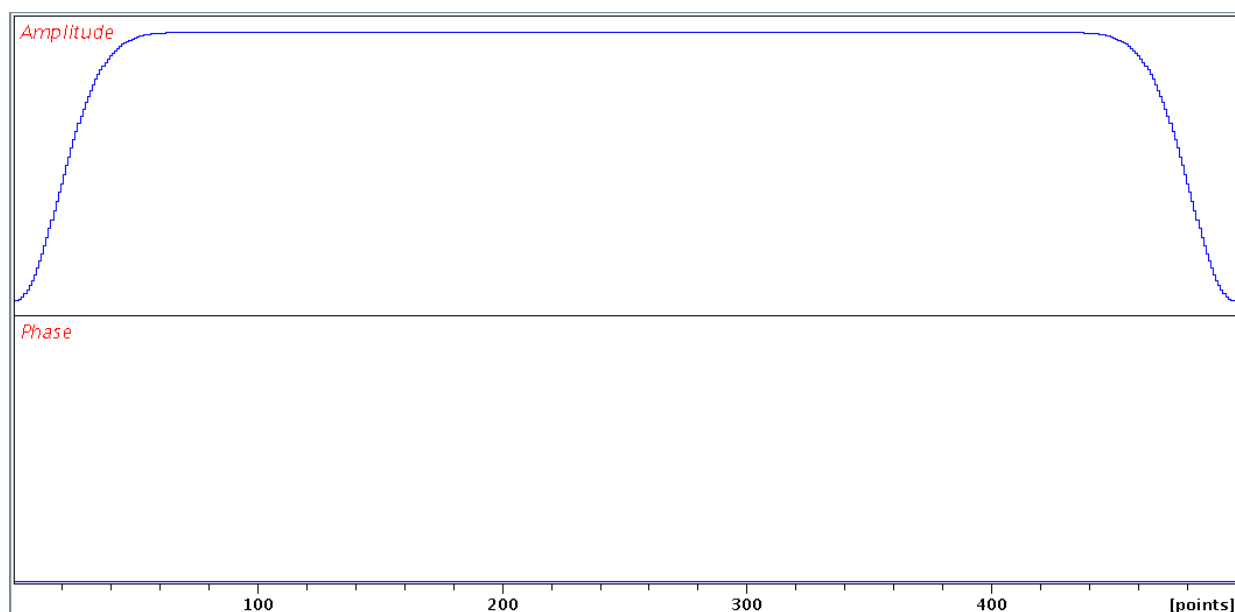


Figure 3.18: Off-resonance WURST-shaped spin lock for ROESY experiments implemented as on-resonance amplitude modulated waveform with no phase ramp.

- `sin2_apodAM` – stretched sine squared spin lock; the steepness of the initial and final stages of the spin lock is controlled by the APOD parameter (in %, must be between 0 and 50), for instance:

```
;sp14:wvm:SL1: sin2_apodAM-10(100 ms; B1max= 5.1 kHz)
```

```
;sp14:wvm:SL1: sin2_apodAM(100 ms; APOD= 10 %, B1max= 5.1 kHz)
```

Both WaveMaker inputs produce equivalent results.

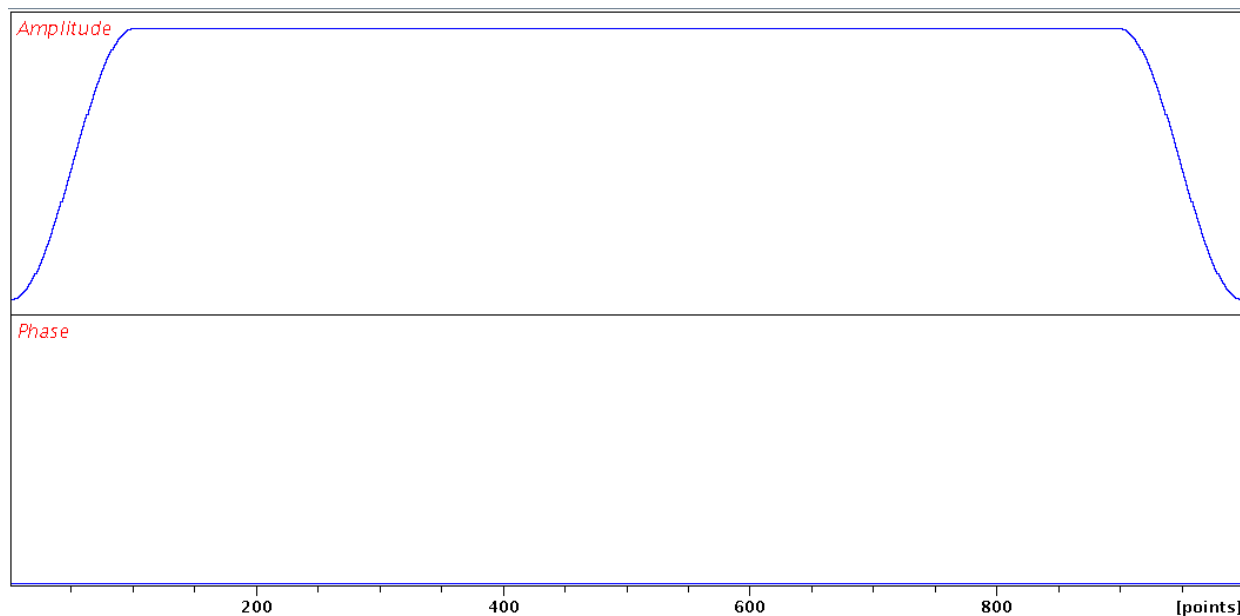


Figure 3.19: Off-resonance sine-squared-apodized shaped spin lock for ROESY experiments implemented as on-resonance amplitude modulated waveform with no phase ramp.

- phase ramped off-resonance spin lock can be implemented as in (a) but instead of a frequency jump the `spoff#` pulse program parameter can be used to initiate an appropriate phase ramp. Alternatively explicit phase ramp can be incorporated into the shape file. In this case the user has control over the phase increments and the number of steps. For example:

```
;sp14:wvm:SL1: wurstAM-80(100 ms, 12 kHz; B1max= 5.2 kHz) phi= 20 deg
```

Similar spin locks can be constructed using Gaussian or hyperbolic secant apodization, as shown in some detail in the next chapter.

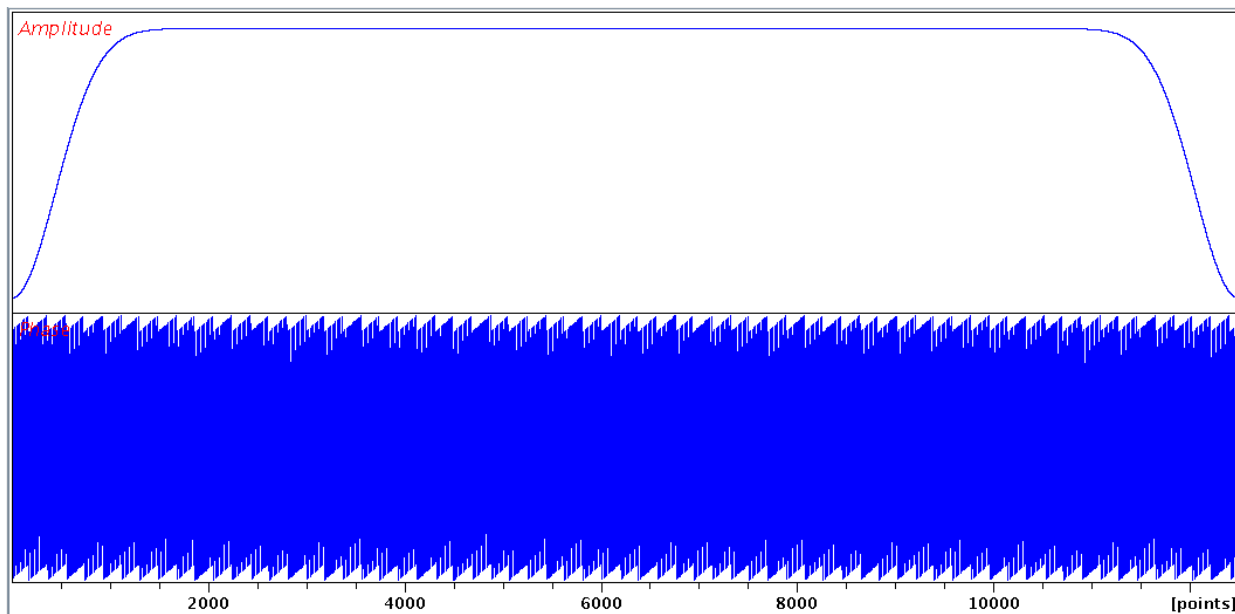


Figure 3.20: Off-resonance wurt-80 shaped spin lock for ROESY experiments implemented as off-resonance waveform with an explicit implementation of the phase-rampe. This implementation requires a significant amount of the waveform memory.

### 3.25 Composite Spinlocks

Some of the off-resonance spin locks that are used in  $T1\rho$  experiments require frequency sweeps (ramps) at the beginning and end of the spin lock wave form. While such spin locks can be created as single off-resonance (phase-ramped) waveforms this type of implementation is usually 'expensive' in terms of the waveform size and the required waveform memory. In order to reduce the size of such waveforms it is customary to jump the carrier frequency to the off-resonance frequency so that the long middle section of the waveform can be executed as an 'on-resonance' shape with constant amplitude (i.e. a rectangular pulse). Assuming the magnetization of interest is aligned with the Z-axis before the spin lock takes place, the most economic implementation of such waveform in the current release of the WaveMaker is a composite pulse consisting of three shapes with the following pulse program element:

```
(p14:sp14 ph0):f1
(p15:sp15 ph0):f1
(p16:sp16 ph0):f1
```

The corresponding WaveMaker definitions are then shown below:

```
; ~~~~~ WaveMaker definitions ~~~~~
;sp14:wvm:SL1: hsech_SL(10 kHz, 2 ms; B1max= 10 kHz)
;sp15:wvm:SL2: cw(100 ms; np=50) B1=-1
;sp16:wvm:SL3: hsech_SL(10 kHz, 2 ms; TR) B1= -2;
```

The resulting waveforms are shown below:

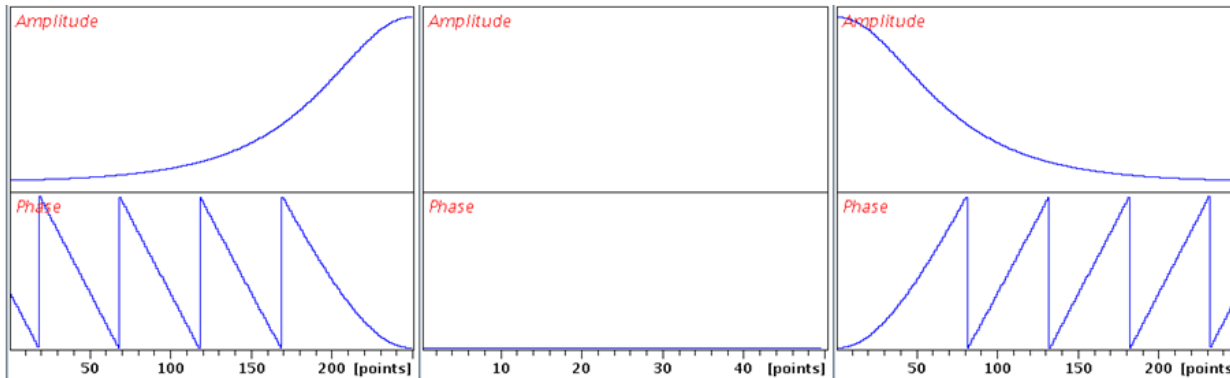


Figure 3.21: An example of a ‘drive-through’ type of adiabatic off-resonance spinlock consisting of hyperbolic secant ramps flanking the main spin lock module. The frequency is swept through the off-resonance offset frequency.

This is a ‘drive through’ option of such a waveform. Clearly the 10 kHz sweep in this particular example may take the final position of the RF field undesirably close to the bandwidth of interest. The user may want to avoid this and adjust the duration and bandwidth of the adiabatic sweeps accordingly, but should take care to preserve the desired properties of the adiabatic sweeps (pw\*bw product of ca. 20, e.g. use 4 kHz 5 ms sweeps). Alternatively one may opt for the ‘out-and-back’ option by changing the sweep direction of the ramp-down pulse so as to sweep away from the active spectral bandwidth. For instance, assuming that the frequency of the spin lock is jumped to a high frequency the WaveMaker instructions would be as below:

```
; ~~~~~ WaveMaker definitions ~~~~~
;sp14:wvm:SL1: hsech_SL(10 kHz, 2 ms; B1max= 10 kHz)
;sp15:wvm:SL2: cw(100 ms; np=50) B1=-1
;sp16:wvm:SL3: hsech_SL(10 kHz, 2 ms; TR, L2H) B1=-2
```

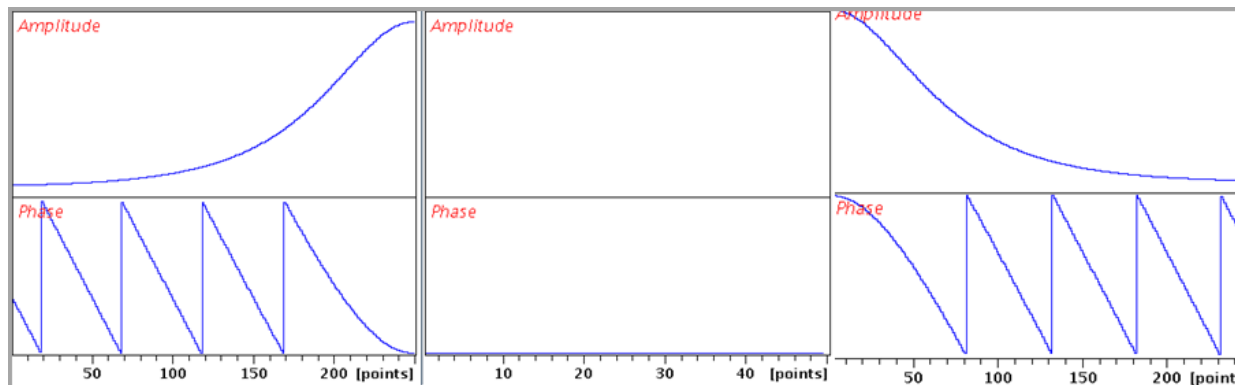


Figure 3.22: An example of a ‘out-and-back’ type of adiabatic off-resonance spin lock consisting of hyperbolic secant ramps flanking the main spinlock module. The frequency is swept through the off-resonance offset frequency.

The sweep direction of the ramp-up pulse is set to ‘H2L’ (high to low field) whereas the sweep direction of the ramp-down pulse is set to ‘L2H’ (low to high field). Note the B1 flag (B1=-1) that carries over the B1max from the previous shape. If not set, the B1 field of SL2 would need a manual adjustment.

If for whatever reason the carrier frequency cannot be jumped, the off-resonance spinlocks must be implemented as frequency modulated waveforms which tend to be long and may consume a lot of waveform memory as indicated above. The WaveMaker provides a facility to compress such waveforms by exploiting the cyclic nature of the phase incrementation, provided the pulse program can be written in the following form:



```
(p14:sp14 ph0):f1
7 (p15:sp15 ph0):f1
lo to 7 times l15
(p16:sp16 ph0):f1
```

In principle, such composite spin lock can also be represented in a CPD program in a very similar way. Assuming an offset of 12 kHz the spin-lock instructions for creating shapes using WaveMaker can then be written as below:

```
; ~~~~~ WaveMaker definitions ~~~~~
;sp14:wvm:SL1: hgaus_SL(2 ms, 12 kHz; B1max= 10 kHz)
;sp15:wvm:SL2: cw(cnst27 ms, 12 kHz) LOOP= L15, B1=-1
;sp16:wvm:SL3: hgaus_SL(2 ms; TR) B1=-1
```

Note that the LOOP flag can be either just activated (LOOP= 1) or set to a particular loop counter, e.g. LOOP= L15. In the latter case the L15 parameter in the current experiment will be set to the required (integer) number. In this example cnst23 represents the duration of active part of the spin lock, e.g. cnst23= 100.0. The resulting waveforms are shown below:

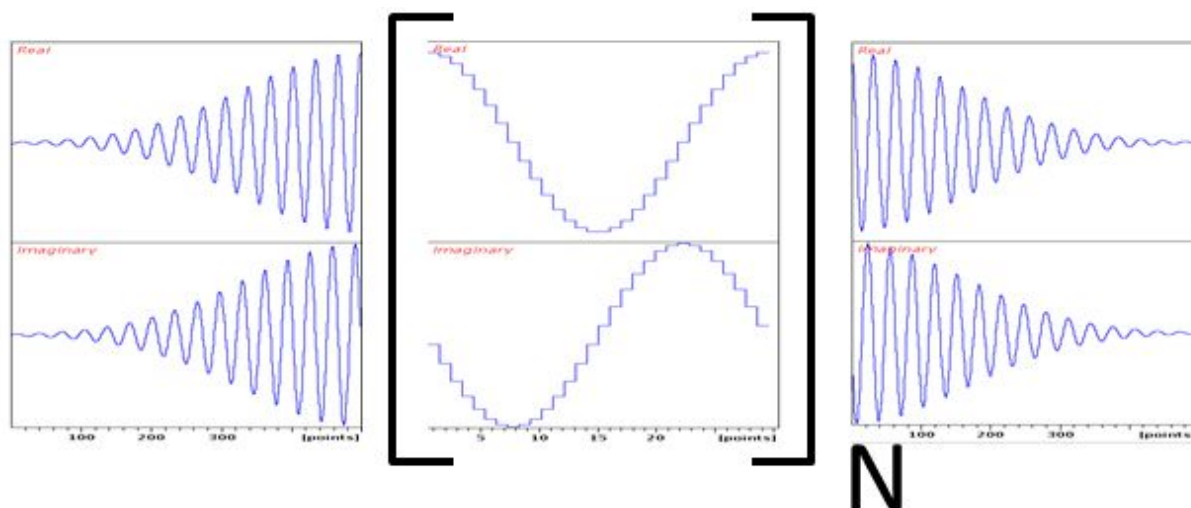


Figure 3.23: An example of a compressed composite off-resonance spin lock consisting of hyperbolic secant ramps flanking the main frequency modulated spin lock module. In this example the frequency is swept through the off-resonance offset frequency.

### 3.26 Spinlocks for Cross-Polarization

Adiabatic cross-polarization in solids is implemented using RF ramps, such as linear ramp (rampAM) or tangential ramp (tanAM) applied to either two or more nuclei simultaneously. Both, the tanAM and rampAM are stored in the shapes/spin locks directory. Again, controls to adjust the steepness of the ramp and the average B1 field are provided. In the case of tanAM sweep the angular constant, C2 is set by default to 80 degrees (C2=80), whereas the constant, C1 that sets the half-bandwidth of the ramp is set to 20 % (C1=20). The direction of the ramp is inverted by using a negative angular constant, e.g. C2=-80. The C3 constant allows for reversing the sweep direction in the middle of the sweep.

The following figures provide examples of tangential amplitude (B1) sweep spin locks for cross-polarization experiments in solids.

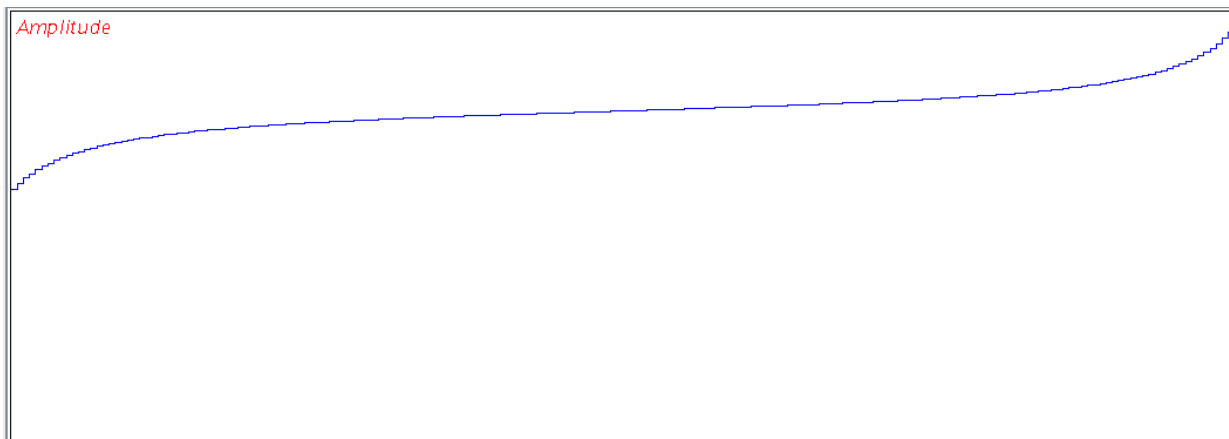


Figure 3.24: Tangential B1 sweep:  $\text{tanAM}(p23, B1av = \text{cnst}21 \text{ kHz})$

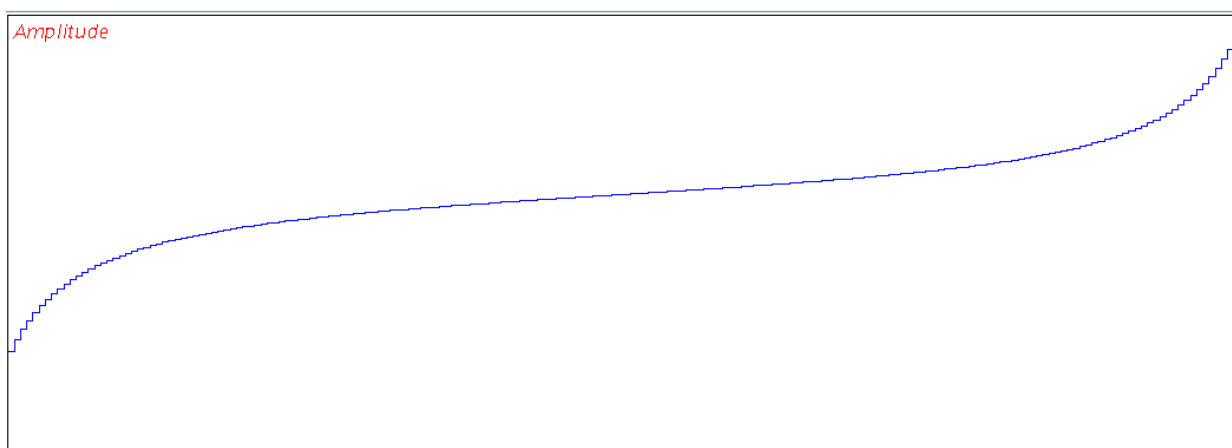


Figure 3.25: Tangential B1 sweep:  $\text{tanAM}(p23, B1av = \text{cnst}21 \text{ kHz}, C1=40)$

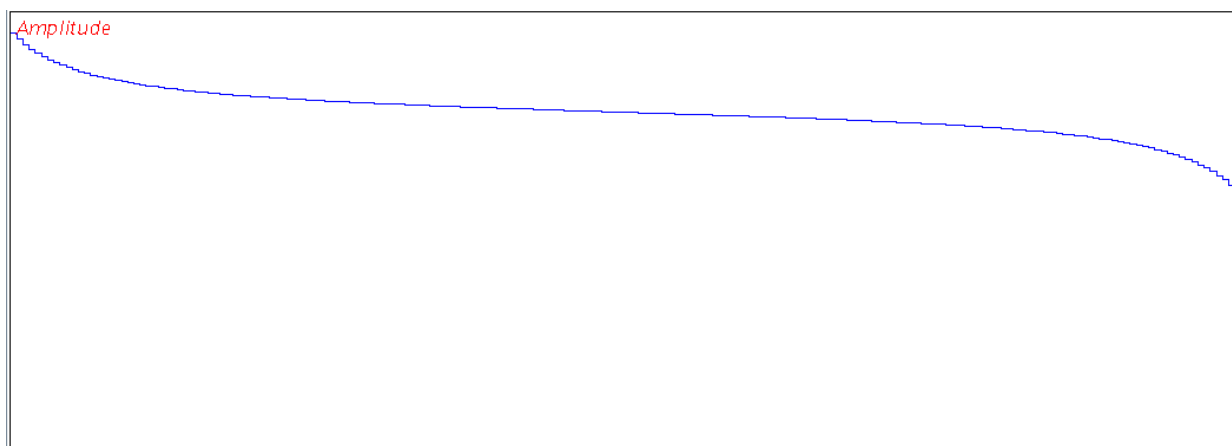


Figure 3.26: Tangential B1 sweep:  $\text{tanAM}(p23, B1av = \text{cnst}21 \text{ kHz}, C2=-80.0)$

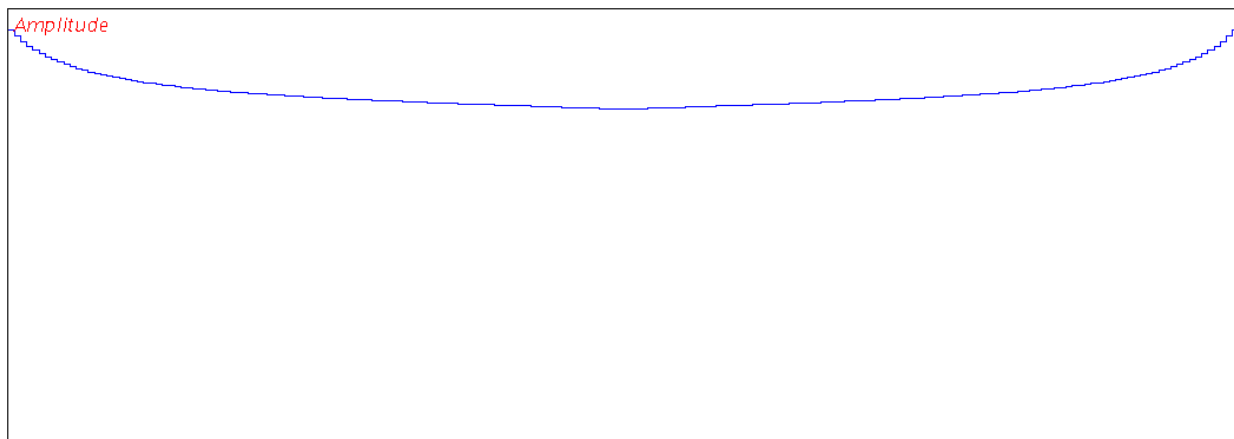


Figure 3.27: Tangential B1 sweep:  $\text{tanAM}(p23, B1av = \text{cnst}21 \text{ kHz}, C3=-1)$

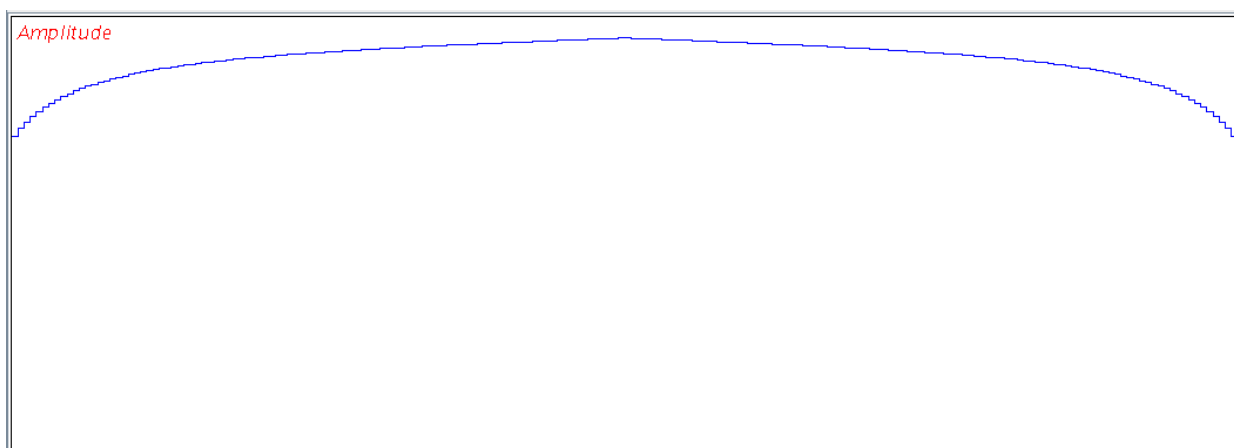


Figure 3.28: Tangential B1 sweep:  $\text{tanAM}(p23, B1av = \text{cnst}21 \text{ kHz}, C3=-1)$

### 3.27 WaveMaker Parameter File

The WaveMaker parameter file is stored in the acquisition directory of the current experiment and may be useful for bug fixing. It consists of three sections: (1) output filename, (2) shape functions and (3) channel parameters, e.g.

```
shp1: eburp1(200.0 Hz, -1.5 kHz)
p90 = 8.2 us
p1 = -7.3 dB
shp2: reburp(200 Hz, -1.5 kHz)
p90 = 8.2 us
p1 = 7.3 dB
```

The name of each output file (waveform) must end with '.'. There is no separator to indicate the end of input for individual waveforms other than a new filename. The waveform file name (wavename) is followed by shape function(s) which in turn is followed by channel specific parameters.

## 4 Appendix

This appendix provides names of the most frequently used shape functions (in arbitrary order).

### Excitation pulse shapes

G4, eburp1, eburp2, esnob, gaus270, qsneeze, sneeze, uburp, Q5, Q5sebop, gaus90, seduce90, square90, cw, evega1, evega5, hgaus90, pc5, sinc90, sinc 90n, square90n, evega2, evega3, evega4, gaus, hsech90, pc9, e0a, e0b, e100a, e100n, e200a, e200d, e200f, e300a, e300c, e300f, e400b, e500a, e500b, e500c, e600c e600f e800a e800b, e1000b;

### Inversion pulse shapes

G3, iburp1, iburp2, isnob2, isnob3, dsnob, cawurst, gaus180, seduce180, sm\_ca\_chirp, square180n, wurst, Q3, chirp, sinc180, sm\_chirp, tanFM, wurst2, sech sinc180n, square180 tanhtan, BIP1, BIP2, BIP3, BIP4, BIP5;

### Refocusing pulse shapes

reburp, rsnob, rvega1, rvega3, seduce180r, sinc180r, square180r, gaus180r, rvega2, rvega4, sinc180nr, square180nr;

### Gradient pulse shapes

ramp\_down\_g, ramp\_up\_g, sin2\_g, sin\_g, square\_g, wurst\_g, ramp\_g, sin2\_apod\_g, sin\_apod\_g, sinc\_g, trapez\_g;

### Decoupling sequences and pulse shapes

waltz16, garp1, garp5, garp16, suzan1, cawurst\_d, chirp\_d, seduce1, sm\_chirp\_d, wurst2\_hd, cawurst\_hd, sm\_ca\_chirp\_d, wurst2\_d, wurst3\_hd;

### Mixing sequences and spin locks

dipsi2, flopsy8, hsech\_SL, rampAM, tanAM, wurstAM, dipsi3, hgaus\_SL, htanh\_SL, sin2\_apodAM, towny;

### Flip-back pulses

G4\_fb, eburp1\_fb, gaus270\_fb, hgaus90\_fb, qsneeze\_fb, sneeze\_fb, uburp\_fb, Q5\_fb, eburp2\_fb, gaus90\_fb, pc5\_fb, sinc90\_fb, square90\_fb, cw\_fb, esnob\_fb, gaus\_fb, pc9\_fb, sinc90n\_fb, square90n\_fb;

### Phase cycles:

comp121, comp90, dp3, m128, m32, m4a, m64, p5, p9, tny, comp180, dp2, g1, m16, m4, m4b, m8, p7, sz1, w16;





**Table 1: WaveMaker input parameters.**

**stepsize** – waveform stepsize in us;  
**phi** – maximum phase increment, determines number of points;  
**npoints** – number of points (np), overruled by phi and stepsize;  
**BF** – base frequency in MHz;  
**BS** – Bloch-Siegert compensation flag, 0 (disabled) or 1 (default, enabled)  
**ofs** – carrier frequency offset in Hz(default), kHz or ppm;  
**p90** – 90 degree pulse duration at power level pl;  
**pl** – power level for p90;  
**nuc** – nucleus in Bruker format, e.g. 1H, 13C, 15N, etc;  
**su** – supercycle, e.g. decoupling phase cycle;  
**B1** – RF amplitude alignment flag; accepted values: 0 (not used), -1 (amplitude set to that of the previous WaveMaker shapefile).  
**LOOP** – loop flag; accepted values: 0 (not used), 1 (active) and L# (active, # from 0 to 32);

**Table 2: Variable Shape Parameters.**

These parameters are typically defined in the /shapes libraries and can be redefined in the WaveMaker input files shape functions; These definitions must be separated from the shape arguments data by a semicolon, ';':

**Q** – adiabaticity;  
**NP** – minimum number of points (np);  
**BW** – bandwidth to pulsewidth product;  
**BWLVL** – the level at which the required bandwidth is set. Forces the WaveMaker to use the Bloch simulator to set the required bandwidth exactly.  
**AM** – amplitude modulation; accepted values are sin (sine) and cos (cosine); these are used to create two symmetric excitation bands around the carrier frequency;  
**B1max** – desired B1 amplitude; accepted values are Hz (default), kHz, W and dB;  
**B1rms** – desired RMS power; accepted values are Hz (default), kHz, W and dB;  
**B1av** – desired average RF amplitude; accepted values are Hz (default), kHz, W and dB;  
**ROT** – rotation angle, deg;  
**TR** – time reversal; accepted values are: no argument (equivalent to TR = 1), 0 (no time reversal) and 1 (time reverse);  
**H2L** – high to low field frequency sweep; set to 1, (active), 0 (ignored) or as a flag;  
**L2H** – low to high field frequency sweep; set to 1, (active), 0 or as a flag;  
**JCOMP** – flag to activate J-compensation for adiabatic pulses; set to 1 (proteins, default) or 2 (RNA/DNA samples);  
**SUCYC** - supercycle (sucyc); change the phase cycle to one of the definitions given in /shapes/phase cycles; nesting can be used, e.g. [[p5]p7]m4 and similar  
**APOD** – apodization factor (in %). Typically used in gradient shapes and off-resonance spin locks. Technically implemented as stretching of shapes such as sin, sin2, triangle, etc at the mid-point. The shapes not necessarily need be symmetric.

**Table 3: Parameters used in /shapes/phase cycles.**

**FUNCT** – function;

**NPOINTS** – number of steps;  
**ROT** – rotation angle;  
**MATRIX** – data matrix;

**Table 4: Parameters used in /shapes libraries.**

**FUNCT** – amplitude function;  
**SWEEP** – frequency sweep function;  
**BDS** – flag for bi-directional frequency sweep; set to 1 (active) or 0 (ignored); Use 1 for cos modulation and 2 for sin modulation;  
**Q** – adiabaticity;  
**BW** – bandwidth to pulse duration product;  
**B1** – gB1 to pulse duration product;  
**ROT** – rotation angle on resonance;  
**C1, C2, C3** – constants;  
**SYM** – symmetry;  
**APOD** – apodization factor (in %). Typically used in gradient shapes and off-resonance spin locks. Technically implemented as stretching of shapes such as sin, sin<sup>2</sup>, triangle, etc. at the mid-point. The shapes not necessarily need be symmetric.  
**NPOINTS** – default number of points;  
**NPmin** – minimum number of points; sets the limit;  
**SUPERCYCLE** – default phase cycle; combination of phase cycles defined in /shapes/phase cycles; the corresponding files must exist;  
**MATRIX** – data matrix;  
**H2L** – high to low field (negative to positive frequency) sweep; set to 1, (active), 0 (ignored) or as a flag;  
**L2H** – low to high field (positive to negative) frequency sweep; set to 1, (active), 0 (ignored) or as a flag;  
**CROP\_H** – crop shapes horizontally; the accepted values are between 0 and 1;  
**CROP\_V** – crop shapes vertically; the accepted values are between 0 and 1;  
**PA** – phase alignment; the accepted values are between 0 and 1;  
**TR** – time reversal; accepted values are: no argument (equivalent to TR = 1), 0 (no time reversal) and 1 (time reverse);

**Table 5: Global WaveMaker parameters.**

These parameters are defined internally and can be modified if necessary, using WVMHOME/wvm\_globals file (optional). The given values are the internally set default values.

-----  
**TShome** = /opt/topspin3.2;  
**pulseassign** = p1p3p21p23  
**powerassign** = pl1pl2pl3pl4  
**NPmax** = 64000 points in TopSpin 3.x and 1e6 points in TopSpin 4.0.7 and later (maximum number of points)  
**SSmin** = 0.050 us (minimum step size)  
**PHImax** = 22.5 deg (maximum phase increment)  
**Gmax** = 53.5 G/cm (maximum gradient strength)



**clockrate** = 10 MHz (clock rate)





## List of Figures

Figure 3.1:	Decoupling sidebands in <sup>13</sup> C HSQC spectra of pamoic acid in DMSO recorded using the standard pulse program, hsqcedetgmsp.3 and CPD program p5m4sp180.p31 (top panel) and bi-level decoupling sequence bi_p5m4sp_4sp.2 (bottom panel) with the decoupling sidebands effectively suppressed in just two scans per t1 increment. ....	12
Figure 3.2:	An example of adiabatic WURST-40 pulse with simultaneous modulation of both, the shape amplitude and frequency (phase). ....	13
Figure 3.3:	Inversion profile of an adiabatic WURST-40 pulse with sweep bandwidth set to 56 kHz and pulse duration to 1.3 ms. Note that the effective inversion bandwidth is only ca 44 kHz. ....	15
Figure 3.4:	An example of a composite adiabatic pulse consisting of three cawurst-20 pulses of total duration set to 2 ms and the sweep bandwidth to 80 kHz as describe in the main text. ....	16
Figure 3.5:	<sup>19</sup> F-spectra of 24 cocktails of 460 compound library from Keyorganics. The ca. 100ppm bandwidth needed for <sup>19</sup> F-experiments corresponds to 66kHz at 700MHz. Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland). ....	17
Figure 3.6:	The experimental and simulated profiles of the adiabatic pulse generated using the shape cawurst-20 (140 ppm, 2 ms) on a 700 MHz instrument. The active bandwidth covers 92 kHz with > 99 % efficiency using 6.95 kHz RF peak amplitude (corresponding to 36 us 90° square pulse). Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland). ...	18
Figure 3.7:	The excitation profile of the squire90n notch pulse showing the special bandwidth definition that indicates the position of the first zero crossing. ....	18
Figure 3.8:	The position of the zero crossing is slightly changing in the pc9 pulses (left panel) depending on the flipangle set to 30, 90 and 120 degrees. In contrast, it is independent of the flip angle in the notch pulse pc9n (right panel). ....	19
Figure 3.9:	Three examples of shifting the origin of one of the shaped pulses in a waveform combining two hyperbolic secant pulses. ....	19
Figure 3.10:	Bloch simulations for double excitation waveforms created using the AM flag and the following shape functions - the left panel: eburp1(0.8 ppm, 3.715 ppm; AM=cos) and the right panel: eburp1(0.8 ppm, 3.715 ppm; AM=sin); In this example o1p = 4.7 ppm and BF1 = 900 MHz. ....	22
Figure 3.11:	The excitation profile of the shape created as described above using E-BURP-1 pulses. ....	23
Figure 3.12:	The effect of BS compensation on double excitation profiles of iburp-2 pulses; Left – with BS compensation disabled (iburp2(3 ppm, 7.6 ppm) iburp2(3 ppm, 3.0 ppm) BS=0) and right – with automatic BS compensation (iburp2(3 ppm, 7.6 ppm) iburp2(3 ppm, 3 ppm)). The offset was set to o1p = 4.7 ppm and BF1 was 900 MHz. ....	23
Figure 3.13:	The effect of time reversal of one of the inversion pulses on the profile of double inversion with iburp-2 pulses. ....	24
Figure 3.14:	The left panel - WaveMaker output in a Hadamard encoding experiment with cnst21=15.0 and the following shape definition: SPL:wvm:xxx: eburp1(cnst21 Hz, PL, had); the right panel – Bloch simulation for the second encoding shape (xxx_1.wv). ....	25
Figure 3.15:	The effect of B1comp flag on artefacts in Hadamard encoding experiment. Spectra are courtesy of Dr. Jihyun Kim and Prof. Lucio Frydman (Weizmann Institute). ....	27
Figure 3.16:	Multi-frequency peak suppression using two cw() waveforms as described in section <a href="#">3.21</a> [ 24]. The RF field strength applied to the EtOH peaks is defined by cnst6 and to the on-resonance H2O peak by cnst7. Figure provided by Dr. Helena Kovacs (Bruker BioSpin AG, Fällanden, Switzerland). ....	28
Figure 3.17:	The homo-decoupling waveform generated by WaveMaker as described in the main text above. ....	28
Figure 3.18:	Off-resonance WURST-shaped spin lock for ROESY experiments implemented as on-resonance amplitude modulated waveform with no phase ramp. ....	29



Figure 3.19:	Off-resonance sine-squared-apodized shaped spin lock for ROESY experiments implemented as on-resonance amplitude modulated waveform with no phase ramp.....	30
Figure 3.20:	Off-resonance wurst-80 shaped spin lock for ROESY experiments implemented as off-resonance waveform with an explicit implementation of the phase-rampe. This implementation requires a significant amount of the waveform memory. ....	31
Figure 3.21:	An example of a 'drive-through' type of adiabatic off-resonance spinlock consisting of hyperbolic secant ramps flanking the main spin lock module. The frequency is swept through the off-resonance offset frequency. ....	32
Figure 3.22:	An example of a 'out-and-back' type of adiabatic off-resonance spin lock consisting of hyperbolic secant ramps flanking the main spinlock module. The frequency is swept through the off-resonance offset frequency. ....	32
Figure 3.23:	An example of a compressed composite off-resonance spin lock consisting of hyperbolic secant ramps flanking the main frequency modulated spin lock module. In this example the frequency is swept through the off-resonance offset frequency. ....	33
Figure 3.24:	Tangential B1 sweep: tanAM(p23, B1av = cnst21 kHz) .....	34
Figure 3.25:	Tangential B1 sweep: tanAM(p23, B1av = cnst21 kHz, C1=40).....	34
Figure 3.26:	Tangential B1 sweep: tanAM(p23, B1av = cnst21 kHz, C2=-80.0) .....	34
Figure 3.27:	Tangential B1 sweep: tanAM(p23, B1av = cnst21 kHz, C3=1).....	35
Figure 3.28:	Tangential B1 sweep: tanAM(p23, B1av = cnst21 kHz, C3=-1) .....	35

## List of Tables

Table 3.1: $B_0$ Field-independent experimental parameters for $J_{CH}$ -compensated pulses. The $J_{max}$ and $J_{min}$ are informative and not used experimentally. ....	14
--	----

---

## 5 Contact

---

### Manufacturer

Bruker BioSpin GmbH  
Silberstreifen 4  
D-76287 Rheinstetten  
Germany

E-Mail: [nmr-support@bruker.com](mailto:nmr-support@bruker.com)

<http://www.bruker.com>

WEEE DE43181702

### Bruker BioSpin Hotlines

Contact our Bruker BioSpin service centers.

Bruker BioSpin provides dedicated hotlines and service centers, so that our specialists can respond as quickly as possible to all your service requests, applications questions, software or technical needs.

Please select the service center or hotline you wish to contact from our list available at:

<https://www.bruker.com/service/information-communication/helpdesk.html>





## Lastpage

<b>1</b>	<b>Introduction.....</b>	<b>5</b>
<b>2</b>	<b>WaveMaker in a Nutshell .....</b>	<b>5</b>
2.1	Waveform Definition in Pulse Programs .....	5
2.2	Shape Functions .....	5
2.3	Shape Specific Parameters .....	6
2.4	Gradient Pulses .....	7
2.5	User defined RF Calibration Parameters .....	7
2.6	Making Shapes .....	7
2.7	Internal Settings .....	8
2.8	ZGOPTNS ignored.....	9
<b>3</b>	<b>WaveMaker Applications .....</b>	<b>9</b>
3.1	WaveMaker Syntax in Detail.....	9
3.2	Absolute vs. Relative Offsets .....	10
3.3	CPD Programs .....	10
3.4	Creating Waveforms for Existing (Standard) CPD Programs - Bi-level Decoupling .....	11
3.5	Gradient Waveforms .....	12
3.6	Adiabatic Pulses .....	13
3.7	J-Compensated Adiabatic Pulses .....	13
3.8	Composite Adiabatic Pulses .....	15
3.9	Adiabatic pulses in <sup>19</sup> F NMR .....	16
3.10	Notch Pulses .....	18
3.11	Optimized Notch Pulses (NEW).....	18
3.12	Time Shifted Pulses (NEW) .....	19
3.13	Shape Definition Files .....	20
3.14	Using TopSpin Parameters.....	20
3.15	Re-defining the Phase Alignment (PA) .....	21
3.16	Re-defining the Excitation Bandwidth .....	21
3.17	Multi-frequency Waveforms .....	21
3.18	Shape Arrays .....	22
3.19	Compensation of Bloch-Siegert Interferences .....	23
3.20	Using Peak Lists and Hadamard Encoding Experiments .....	24
3.21	Multiple (Solvent) Peak Suppression .....	27
3.22	Adiabatic Homo-decoupling .....	28
3.23	Specifying the Required B1max, B1rms and B1av .....	29
3.24	Off-resonance Spin Locks.....	29
3.25	Composite Spinlocks .....	31
3.26	Spinlocks for Cross-Polarization .....	33
3.27	WaveMaker Parameter File .....	35
<b>4</b>	<b>Appendix .....</b>	<b>35</b>
	<b>List of Figures.....</b>	<b>41</b>
	<b>List of Tables .....</b>	<b>43</b>



5 Contact ..... 43

 **Bruker Corporation**

[info@bruker.com](mailto:info@bruker.com)  
[www.bruker.com](http://www.bruker.com)



Order No: H164492