



## Nesting Control Intervals in AcqNmr Pulse Sequences

Ing.Dr.Stanislav Sykora, August 28, 2004

### Introduction

Occasionally, Stelar Users need to insert/attach one or more intervals into/to a particular pulse sequence interval. The most common reasons why such a need arises are:

1. **Generating trigger pulse(s)** for an external device (e.g., a scope)
2. **Generating TTL control sequences for an external device.**

In AcqNmr, the goal is achieved by scripting the sequences and inserting the scripts either into the default sequences library (the *DefaultFfcSequences.ssf* file) or into a User sequences file (see the run-time AcqNmr manual Chapter on *Advanced Software Topics | Pulse sequence Scripts*).

In practice, most such sequences are variations on one of the existing sequences in which one of the intervals is replaced by a more complex sequence of intervals. In this case, the easiest way of generating the new sequence is copying the script of the basic sequence, assigning a new name to the new sequence and modifying the sequence itself to reflect the new requirements.

Sequence scripting rules are described in the run-time manual Chapter *Advanced Software Topics | Pulse Sequence Scripts*. However, the practical example presented in this Note probably illustrates more effectively how to face the relatively simple task at hand (it originates from a User's specific request).

In our example we shall take one of the basic FFC sequences, namely the thermally balanced PP/S sequence, and replace its TAU interval with the following sub-sequence of four intervals:

1. Tau2 interval during which to switch ON the pulser channel **O**.
2. Tau3 interval during which the **O**-channel is OFF.
3. Another Tau2 interval with the **O**-channel ON.
4. Tau4 interval with the **O**-channel is OFF.

During all four intervals, the magnet is to be kept at the relaxation field by means of the standard option switch  $m(R)$ . The **O**-channel (AQM connector labeled P6) is going to be controlled by the option switch  $c(O)$  as described in the Application Note AN040824 (Direct Pulser-Channels Control in AcqNmr Pulse Sequences).

With respect to the original sequence, we will need two extra delay interval parameters for Tau2 and Tau3. We can re-use the delay interval parameter originally assigned to TAU for the new sequence's Tau4 interval.

Once we will have the modified sequence, it shall be possible to tailor it further in different ways, depending upon which interval is to be varied during a multi-block sequence and upon eventual imposed constraints (such as whether the total duration of the four intervals should be kept constant or not). This is done by specifying the interval duration's either as absolute values or as expressions.

We will consider two examples of such special tailoring of the same basic sequence:

- I. Tau2 and Tau4 are constant, while Tau3 is stepped through a multi-block range (if desired, the Tau4 duration may be set to zero).
- II. Tau2 is constant and Tau3 is stepped through a range, but Tau4 is adjusted in a way to keep the sum  $\text{Tau2} + \text{Tau3} + \text{Tau2} + \text{Tau4}$  equal to a constant TAU value.

### Step 1: Copying and renaming the original sequence

To start with, open the library file *DefaultFfcSequences.ssf* with WordPad or another plain-text editor (not Word!) and search for the line starting with

```
SEQUENCE PP/S
```

Now select the block starting with this line and ending with

```
END SEQUENCE
```

and copy it to just after its last line or, if you wish, to the end of the file. You could also copy it into your own sequences library file, avoiding modifications to the default library (it changes with every software update!). For more details, see the run-time manual Chapter *Advanced Software Topics |Pulse sequence Scripts*.

Once you have made the copy of the sequence, modify the first line to something like

```
SEQUENCE PP/S_Tau2324_CaseI
```

Now you have got two distinctly named sequences which shall be both automatically recognized by the AcqNmr software. So far, however, they do exactly the same things.

For comparison, we show here the listing of the original sequence (without its header comments):

```
SEQUENCE PP/S
D0;D1;D2;D3;D4;D5;D6;PW;RINH;ACQD;STIM
#SETUP
D0 :(RD) [m(O)] 'Recycle delay
D1 :(4*T1MX-TAU)[m(R)] 'Balancing
D2 =(SWT) [m(P)] 'Switching to pol.field
D3 :(TPOL)[m(P)] 'Polarization
D4 =(SWT) [m(R)] 'Switching to rlx.field
D5 :(TAU) [m(R)] 'Relaxation interval tau
D6 =(SWT) [m(A)] 'Switching to acq.field
PW = 90 [m(A) p(x,-x,y,-y)]
RINH [m(A)]
ACQD [m(A)]
STIM [m(A)]
ENDS [m(O) p(x,-x,y,-y)] 'Accumulation
'-----
TPOL=.1
BRLX 0.1
'-----
NBLK= 1 'Multi-blocks
BACR= "TAU"
BINI= (4*T1MX)
NBLK= 16
END SEQUENCE
```

### Step 2: Modifying the sequence

The original sequence uses delays D0 to D6. Since we need two more, we must introduce D7 for Tau2 and D8 for Tau3. The basic modified sequence will therefore be (modified/new lines are in **bold red**):

```
SEQUENCE PP/S_Tau2324_CaseI
D0;D1;D2;D3;D4;D7;D8;D7;D5;D6;PW;RINH;ACQD;STIM
#SETUP
D0 :(RD) [m(O)] 'Recycle delay
D1 :(4*T1MX-TAU)[m(R)] 'Balancing
D2 =(SWT) [m(P)] 'Switching to pol.field
D3 :(TPOL)[m(P)] 'Polarization
D4 =(SWT) [m(R)] 'Switching to rlx.field
D5 = 0 [m(R)] 'Tau4
D6 =(SWT) [m(A)] 'Switching to acq.field
D7 = 1e-3 [m(R) c(O)] 'Tau2
D8 :(TAU) [m(R)] 'Tau3
PW = 90 [m(A) p(x,-x,y,-y)]
RINH [m(A)]
ACQD [m(A)]
STIM [m(A)]
ENDS [m(O) p(x,-x,y,-y)] 'Accumulation
'-----
TPOL=.1
```

```

BRLX 0.1
'-----
NBLK= 1           'Multi-blocks
BACR= "TAU"
BINI= (4*T1MX)
NBLK= 16
END SEQUENCE

```

With these few self-evident modifications, we have obtained the modified sequence in which Tau2 (D7) and Tau4 (D5) are accessible to User editing (the = sign after the parameter setup spec). Tau3 (D8) is accessible indirectly through the parameter TAU so that direct User editing can be inhibited (the : sign after the parameter setup spec). The TAU parameter is also used as the one to vary in the multi-block arrayed-parameter experiment (the line BACR="TAU"). Default values are set to 1ms for Tau2 and 0 for Tau4, while the range of Tu3 values (alias TAU) is given by the multi-block specifications for BINI, BEND and NBLK.

This corresponds to our specially-tailored case I.

### Step 3: Special tailoring

We shall now modify the parameter values to make the sequence compatible with the specially-tailored case II (bold blue indicates differences between this case and the previous one):

```

SEQUENCE PP/S_Tau2324_CaseII
D0;D1;D2;D3;D4;D7;D8;D7;D5;D6;PW;RINH;ACQD;STIM
#SETUP
D0 :(RD) [m(O)] 'Recycle delay
D1 :(4*T1MX-TAU)[m(R)] 'Balancing
D2 =(SWT) [m(P)] 'Switching to pol.field
D3 :(TPOL)[m(P)] 'Polarization
D4 =(SWT) [m(R)] 'Switching to rlx.field
D5 =(TAU-2*D7-D8) [m(R)] 'Tau4
D6 =(SWT) [m(A)] 'Switching to acq.field
D7 = 1e-3 [m(R) c(O)] 'Tau2
D8 : 0 [m(R)] 'Tau3
PW = 90 [m(A) p(x,-x,y,-y)]
RINH [m(A)]
ACQD [m(A)]
STIM [m(A)]
ENDS [m(O) p(x,-x,y,-y)] 'Accumulation
'-----
TPOL=.1
BRLX 0.1
'-----
NBLK= 1           'Multi-blocks
BACR= "D8"
BINI= (4*T1MX)
NBLK= 16
END SEQUENCE

```

The modifications may look small but they have quite an impact on the sequence's behavior. According to the specification BACR="D8", the varying parameter of the multi-block sequence is now D8, which is still our Tau3. The starting value of D8 is therefore irrelevant since it will be overwritten by all the arrayed values; we have set it to 0. As before, D7 is a constant, User-definable parameter, but D5 is now constrained by the formula to the value of (TAU-2\*D7-D8). This confers a special meaning to the User definable parameter TAU which now stands for the total duration of Tau2+Tau3+Tau2+Tau4 and is kept constant for all the multi-block sub-experiments.

It is up to the User to make sure that D5=(TAU-2\*D7-D8) never becomes negative. Since the smallest of these arrayed values is in this example TAU-2\*D7-4\*T1MX, the condition amounts to TAU > 4\*T1MX+2\*D7. If it is not satisfied, every attempt at pulser programming will generate the error message

"*Negative D5 ... Parameters incompatibility*" and the pulser programming shall be aborted (no way to run the experiment until the constraint-violating condition is removed by incrementing TAU).

### Conclusions

We hope that these examples have convinced you that inserting control sub-sequences into pre-existent sequence scripts is very simple. The insertions could be as simple as a single added interval or quite complex with many distinct pulses, using a number of external pulser channels in various combinations.

In our example, since only the interval originally linked to TAU is modified, it makes no difference what happens before or after. Consequently, the modification can be applied to any relaxation-type sequence, regardless of its polarization preamble (PP or NP, Balanced or Unbalanced, IR, ...) and of its detection mode (FID, Echo, CPMG, XRing, ...). With two new specially tailored variants, the result would be tripling the number of existing sequences.

It is definitely impossible for Stelar to generate, document and insert all such "new" sequences into the default sequences library. Doing so every time a User requests a new variation, the total number of sequences would soon exceed millions (this is not a joke: if every variation tripled the existing sequences, 13 variations would multiply their number by 1.5 millions). Rather, Users *must* understand the versatility of the software and *the ease with which they can implement such variations themselves* in a matter of minutes.

Stelar encourages requests for explanations of various aspects of writing/modifying NMR and FFC measurement sequences. Such explanations will generally take the form of Application Notes with commented examples. Other forms (like training courses during User meetings) will also be considered, but dedicated work on specific single-User requests should be limited to those rare cases which require modifications of the executable files. Otherwise, since the "space" of all possible sequence variations of potential interest is truly infinite, implementing all of them would soon take up all our time.