



## User extensions of AcqNmr parameter capabilities: Parameter settings by means of formulae

Ing.Dr.Stanislav Sykora

### Introduction

Some Stelar Users have asked for the possibility to define the value of a numeric AcqNmr parameter indirectly by means of an expression involving another parameter(s). A typical example might be setting the T1MX parameter as a function of the relaxation field BRLX.

The AcqNmr software intrinsically supports the concept and *parameter formulae* are widely used in the default library sequences to calculate/adjust the duration's of *pulser intervals*.

**For safety reasons**, the applicability of the formulae is limited to pulser interval parameters, such as various delays and pulse widths.

In some cases, however, the restrictions may appear excessive and/or undesirable. When it is the case, they can be easily removed by a knowledgeable User (it is then up to the User to make sure that no excessive magnet or RF transmitter heating can occur).

A numeric parameter's capability to take part of algebraic formulae is defined by specific letter-flags in its definition script located in the ASCII file *DefaultParameters.def*.

The pertinent flags are:

C ... Parameter may be computed through a formula

V ... Parameter's acronym is admissible as an operand (variable) in expressions

Another flag which we shall find of interest in this context is

p ... Parameter affects pulser timing.

For more details, consult the following sections of the runtime AcqNmr manual:

Chapter *Basic software\System parameters*:

(1) Parameter value formulae:

*Editing of parameters\Values of parameters*  
(or search the index for *Values of parameters*)

(2) Pulser interval parameters:

*Categories of parameters\System interval parameters*  
(or search the index for *System interval parameters*)

In addition, you might wish to consult the Chapter *Advanced software topics\Parameters definitions file* for the description of parameter definition scripts, including all the parameter flags:

### Relaxing the imposed constraints

In our example, the parameter definition entries for T1MX and BRLX in the *DefaultParameters.def* file are, respectively (stripping off the comment field):

```
T1MX|Maximum T1          |0| 0| 11|HDWUpVF |.1  |R |10e-6,1000
BRLX|Relaxation field    |6| 3| 17|ADWUF   |.1  |R |0,1000 |
```

Since T1MX does not have the letter C set among its letter flags, it can not be computed through a formula (though, having the V flag, it may appear as argument in expressions, including formulae for other parameters). BRLX has neither the C flag nor the V flag so that it can not be computed, nor can it appear as an expression argument. These default settings prevent us from setting T1MX to, for example,

$$T1MX = (0.3/(1+BRLX*BRLX))$$

because that implies BRLX as an argument and T1MX as a computed value.

If we want to relax these two constraints, we must modify the definition scripts of the two parameters as follows (modifications are in bold red typeface):

```
T1MX|Maximum T1          |0| 0| 11|HDWUpVFC |.1  |R |10e-6,1000
BRLX|Relaxation field    |6| 3| 17|ADWUFVp  |.1  |R |0,1000 |
```

Now, once we have saved the modified *DefaultParameters.def* file and restarted AcqNmr, the desired setting for T1MX shall be accepted and become operative.

Notice that we have added the p-flag to BRLX. The reason for doing this is to tell the AcqNmr software that whenever BRLX is edited, the pulse sequence timing changes and therefore the pulser needs to be updated. Without this flag, changing BRLX would not automatically update the pulser (normally, that is not required) and therefore the T1MX value would not change (parameter formulae are evaluated only while the pulser is being programmed).

Other details one should keep in mind:

- Please, always **keep a backup copy of the original *DefaultParameters.def* file !!!**
- Use a plain text Editor to make the modifications (for example, NotePad or WordPad).
- The parameter flag letters are case-sensitive (capital C and V, small p)
- The order in which the letter flags appear in the script is irrelevant
- Do not forget the outer parentheses when setting a parameter value to an expression;  
 $T1MX = 0.3/(1+BRLX*BRLX)$  would not work; it would always default to  $T1MX = 0.3$ .

### Formulae chaining

Formulae can be evaluated in consecutive steps using intermediate expressions. For this purpose one can use the auxiliary parameters AUX1,...,AUX4 (actually, there are 9 of them but only the first 4 are displayed in the *AcqPar|User* parameters page). All the Auxiliary parameters have the C and V flags set so that they may be computed as well as used as arguments in formulae. For example, setting

$$AUX1 = 0.03$$

$$AUX2 = (AUX1*BRLX)$$

$$T1MX = (0.2/(1+AUX2*AUX2))$$

is equivalent to

$$T1MX = (0.2/(1+(0.03*BRLX)*(0.03*BRLX))),$$

making it much easier for the User to change the value of the coefficient 0.03 without having to retype the whole formula.

### Expression restrictions

In AcqNmr versions **up to 2.1.0.55**, expressions accept only the four basic arithmetic operators +, -, \*, / and constant values or parameter acronyms as operands (provided the parameters have the V flag set). There is no limit to parentheses nesting and/or formulae chaining.

Since up to now it had not been requested, they do not support functions (such as exp, log, ...). This is likely to change in future versions. At present, complicated functional dependencies should be modeled using polynomial or rational approximations.

### Persistence of the modifications

In our example, the enhanced capabilities of the T1MX and BRLX parameters are given by their definition scripts and therefore remain in effect as long as the modified *DefaultParameters.def* file is present in the AcqNmr program directory (the one where the AcqNmr.exe file resides).

The settings of parameters through formulae are persistent across sessions. They can be also saved/reloaded to/from parameter files and they are also recorded in the data file headers so that they reappear when you load parameters from a data file zone.