Supporting Information

A new class of CEST experiment based on selecting different magnetization components at the start and end of the CEST relaxation element: an application to ^{1}H CEST

Tairan Yuwen[1] and Lewis E. Kay[1,2]

[1]Departments of Molecular Genetics, Biochemistry and Chemistry, University of Toronto, Toronto, Ontario, M5S 1A8, Canada

[2]The Hospital for Sick Children Research Institute, Program in Molecular Medicine, Toronto, Ontario, M5G 0A4, Canada

Table of Contents

Pulse Sequence Code (Bruker) **3**

TROSY scheme (backbone NH sites) 3

Non-TROSY scheme (sidechain CH_3 sites) 14

```
/* N15_1H_CEST_trosy_lek_800_cp

  This pulse sequence will allow one to perform the following experiment:
  1H CEST when magnetization is of the form Iz with 15N-1H HSQC readout.

  Pervusion TROSY based hsqc including a flag for gradient coherence transfer selectio
n

  Note: Identical trosy selection can be achieved by inverting the phase of the first
1H 90 after t1
        along with the final 15N 90 and then inverting the first 2 or last 2 receiver
phases
        Addition of coherence transfer gradients does not change the cycle

  If gd_sel is used then the min phase cycle is 2
  If gd_sel is not used then the min phase cycle is 4

  To callibrate B1 field: set -Dcal_HB1
    - calibrate by setting 1H decoupling field to 0
    - sit on one of the 1H multiplet components and evolution will be of the other com
ponent
    - calibration is done with magnetization Iz at the very beginning

  Used with N15cest_make_fq_list.c (ccode) which makes the frequency list
  Used with N15cest_cal_B1.c (ccode)
  Used with move_lek_cest_fq which moves the freq list to f1
  Used with move_lek_cest_B1 which moves the list of times to vd list

  Hsteady_flg is for measuring the decay of Hz magnetization to extract R1 rates

  Written from 1H_CEST_Iz_trosy_lek_600_cp on Sept 26(!) 2016 by LEK

  Modified based on 1H_CEST_Iz_TRsss_ty_800_cpz by TY on Aug 18th, 2017.
  This new version detects HzNz magnetization after CEST period.

*/

;prosol relations=<triple>
;aqseq 321

#include <Avance.incl>
#include <Grad.incl>
#include <Delay.incl>

/**********************/
/*   Define phases    */
/**********************/
#define zero ph=0.0
#define one ph=90.0
#define two ph=180.0
#define three ph=270.0

/*************************/
/*    Define delays      */
/*************************/
define delay hscuba      /* length of 1/2 scuba delay */
  "hscuba=30m"
define delay taua        /* 1 / 4J(XH)                */
  "taua=d3"              /* 1s/(4*105) use JNH=105 to decrease the tauxh duration */
define delay BigT1
  "BigT1=d14"
define delay BigT
  "BigT=d15"

#if !defined(cal_HB1) && !defined(cal_HR1)
  define delay time_T1
    "time_T1=d17"        /* CEST duration - typically 100-500 ms */
#ifdef eta_flg
  define delay taud
```

```
    "taud=d1-time_T1"

  "l5=larger(l4,1)"
  "l6 = (trunc(l5 + 0.2))"

  define delay taue
    "taue=(time_T1*0.5)/l6"
#endif
#endif

"d11=30m"
"in0=inf1/2"             /* t1/2 increment */

/*******************************************************************/
/* f1180: Start t1 at half dwell to get -90/180 phase correction */
/*        in F1 (15N) dim - set zgoptns -Df1180                  */
/*******************************************************************/
#if defined(f1180) && !defined(cal_HB1)
  "d0=(in0)/2"
#else
  "d0=0.1u"
#endif

/**************************/
/*    Define pulses      */
/**************************/
define pulse dly_pg1     /* Messrle purge pulse */
  "dly_pg1=5m"
define pulse dly_pg2     /* Messrle purge pulse */
  "dly_pg2=dly_pg1/1.62"
define pulse pwh
  "pwh=p1"               /* 1H hard pulse at power level pl1 (tpwr) */
define pulse pw_sl
  "pw_sl=p13"            /* 1H water selective pulse at power level pl13 (tpwrsl) */

#if defined(Hsteady_flg) || !defined(gd_sel)
  define pulse pw_sl1
    "pw_sl1=p14"         /* 1H water selective pulse at power level pl14 (tpwrsl1) */
#endif

define pulse pwn
  "pwn=p3"               /* PW90 for N pulse at power level pl3 (dhpwr2)  */

#ifdef c_flg
  define pulse pwc_ad    /* 13C adiabatic pulse at power level spw22(pwc_ad) */
    "pwc_ad=p22"
#endif

#ifdef N_sel
  define pulse pwn_sl
    "pwn_sl=p32"
#endif

/**********************************/
/*    Define other parameters    */
/**********************************/
define loopcounter Nphase
  "Nphase=1"

#ifdef cal_HB1
  define list<delay> time_cal=<t1list_calB1>
#else
#ifdef cal_HR1
  define list<delay> time_cal=<t1list_calR1>
#else
  define list<frequency> H_offset=<$FQ1LIST>
#endif
#endif
```

```
/****************************************************/
/* Assign cnsts to check validity of parameter range */
/****************************************************/
#ifdef fsat
  "cnst10=plw10"         /* tsatpwr - set max at 0.00005W */
#endif

#ifdef mess_flg
  "cnst11=plw11"         /* tpwrmess pl11 - set max at 2.0W */
#endif

  "cnst13=spw13"         /* tpwrsl - set max at 0.001W" */

#if defined(Hsteady_flg) || !defined(gd_sel)
  "cnst14=spw14"         /* tpwrsl1 - set max at 0.001W" */
#endif

#ifdef c_flg
  "cnst22=spw22"         /* d_ad - set max at 57W */
#endif

#ifdef N_sel
  "cnst32=spw32"         /* power level for 15N selective pulse */
#endif

/************************************************/
/* Set the weak H-amide B1 field for irradiation */
/************************************************/
"p15=1000000.0/(4.0*cnst16)"
"plw15=plw1*(pwh)*(pwh)/((p15)*(p15))"
"cnst15=plw15"

/****************************************************/
/* Set phase alignment and offset for shaped pulses  */
/****************************************************/
"spoal13=0"
"spoal14=0"
"spoff13=0"
"spoff14=0"

/*****************************/
/* Initialize loop counters */
/*****************************/
"l2=0"
"l3=1"

;"acqt0=0"
;baseopt_echo

1 ze
/******************************************/
/*   Check Validity of Parameter Range    */
/******************************************/
#ifdef fsat
  if "cnst10 > 0.00005" {
    2u
    print "error: tsatpwr pl10 too large !!!  "
    goto HaltAcqu
  }
#endif

#ifdef mess_flg
  if "cnst11 > 2.0" {
    2u
    print "error:tpwrmess pl11 too large !!!  "
    goto HaltAcqu
  }
#endif
```

```
  if "cnst13 > 0.002" {
    2u
    print "error: tpwrsl spw13 too large !!!  "
    goto HaltAcqu
  }

#if defined(Hsteady_flg) || !defined(gd_sel)
  if "cnst14 > 0.002" {
    2u
    print "error: tpwrsl1 spw14 too large !!!  "
    goto HaltAcqu
  }
#endif

  if "cnst15 > 0.01" {
    2u
    print "error: cnst15 - power in W for 1H CEST is too large !!!  "
    goto HaltAcqu
  }

  if "cnst16 > 250" {
    2u
    print "error: cnst16 - weak 1H B1 field strength in Hz must be <= 250 Hz !!!  "
    goto HaltAcqu
  }

#ifdef c_flg
  if "cnst22 > 50" {
    2u
    print "error: pwc_ad spw22 is too large !!!  "
    goto HaltAcqu
  }
#endif

#ifdef N_sel
  if "cnst32 > 35" {
    2u
    print "error: spw23 is too large < 55W !!!  "
    goto HaltAcqu
  }
#endif

#if !defined(cal_HB1) && !defined(cal_HR1)
  if "time_T1 > 0.5s" {
    2u
    print "error: time_T1 is too long"
    goto HaltAcqu
  }
#endif

#ifdef eta_flg
  if "time_T1 > d1" {
    2u
    print "error: d1 should be longer than time_T1"
    goto HaltAcqu
  }
#endif

2 d11
/*****************************************/
/* Continue to check real time variables */
/*****************************************/
#ifdef cal_HB1
  "DELTA=time_cal[l2]"
  if "DELTA > 500m" {
    2u
    print "error: time_cal (evolution time during B1 calib is too long < 500 m"
    goto  HaltAcqu
  }
```

```
#endif

/**************************************************/
/*   Presaturation Period                         */
/*   option for Messrlie purge zgoptn -Dmess_flg  */
/**************************************************/
#ifdef mess_flg
  20u pl11:f1
  (dly_pg1 ph26):f1
  20u
  (dly_pg2 ph27):f1
  20u pl10:f1
#endif /*mess_flg*/

#ifdef fsat
  4u pl10:f1
#if !defined(eta_flg) || defined(cal_HB1) || defined(cal_HR1)
  d1 cw:f1 ph26
#else
  taud cw:f1 ph26
#endif
  4u do:f1
  2u pl1:f1
#ifdef fscuba
  hscuba
  (pwh ph26):f1
  (pwh*2 ph27):f1
  (pwh ph26):f1
  hscuba
#endif /*fscuba*/
#else
  2u pl1:f1
#if !defined(eta_flg) || defined(cal_HB1) || defined(cal_HR1)
  d1
#else
  taud
#endif
#endif /*fsat*/

  20u UNBLKGRAD

/****************************************/
/* Erase 15N equilibrium magnetization */
/****************************************/
  2u pl3:f3
  (pwn ph26):f3

  2u
  p51:gp1
  d16

/**************************************************/
/*  +/- phase cycle on 1H prior to CEST period    */
/**************************************************/
#ifdef Hsteady_flg
  2u
  (pw_sl1:sp14 ph27):f1
  2u pl1:f1
  (pwh ph26 pwh*2 ph21 pwh ph26):f1
  2u
  (pw_sl:sp13 ph22):f1

  2u
  p51:gp1
  d16
#endif /*Hsteady_flg*/

/*****************************************/
/*   CEST portion on Hz magnetization   */
```

```
/****************************************/
  2u pl15:f1
#ifdef cal_HB1
  "DELTA=time_cal[l2]"
  2u fq=cnst1:f1
  DELTA cw:f1 ph26
  2u do:f1
#else
#ifdef cal_HR1
  "DELTA=time_cal[l2]"
  DELTA
#else
  2u fq=H_offset:f1
#ifdef eta_flg
4 taue cpds1:f1 ph26
  2u do:f1

  (pwn ph27 pwn*2.0 ph26 pwn ph27):f3
  taue
  taue
  (pwn ph27 pwn*2.0 ph26 pwn ph27):f3

  taue cpds1:f1 ph26
  2u do:f1
  lo to 4 times l6
#else /*eta_flg*/
  time_T1 cw:f1 ph26
  2u do:f1
#endif /*eta_flg*/
#endif /*cal_HR1*/
#endif /*cal_HB1*/
  20u fq=0:f1 pl1:f1

  2u
  p52:gp2
  d16

/*****************************************/
/* Special setup for the reference plane */
/*****************************************/
#ifndef cal_HB1
#ifdef cal_HR1
  "l3=0"
#else
  if "abs(H_offset)*1e6 > 10000" {
    "l3=0"
  }
  else {
    "l3=1"
  }
#endif /*cal_HR1*/
#endif /*cal_HB1*/

  if "l3 == 0" {
    (pwh ph26):f1

    2u
    p54:gp4
    d16

#ifdef N_sel
    "DELTA = taua - 2u - p54 - d16 - pwn_sl*0.5"
#else
    "DELTA = taua - 2u - p54 - d16"
#endif
    DELTA

#ifdef N_sel
    (center (pwh*2 ph26):f1 (pwn_sl:sp32 ph26):f3)
```

```
#else
    (center (pwh*2 ph26):f1 (pwn*2 ph26):f3)
#endif

    DELTA

    2u
    p54:gp4
    d16

    (pwh ph27):f1

    /* shaped pulse */
    4u
    (pw_sl:sp13 ph26):f1
    2u pl1:f1
    /* end shaped pulse */
  }
  0.2u

  2u pl3:f3
  p55:gp5
  d16

/*********************/
/*   15N labeling    */
/*********************/
#ifndef Hsteady_flg
  if "Nphase %2 == 1" {
    (pwn ph1):f3
  }
  else {
    (pwn ph2):f3
  }
#else
  if "Nphase %2 == 1" {
    (pwn ph11):f3
  }
  else {
    (pwn ph12):f3
  }
#endif /*Hsteady_flg*/

#ifdef c_flg
  if "d0 - pwc_ad*0.5 > 2u" {
    "DELTA = d0 - pwc_ad*0.5"
    DELTA
    (pwc_ad:sp22 ph26):f2
    DELTA
  }
  else {
    d0
    d0
  }
#else
  d0
  d0
#endif /*c_flg*/

#ifdef gd_sel
  2u
  p56:gp6*-1.0
  d16

  "DELTA = BigT - 2u - p56 - d16"
  DELTA

  (pwn*2 ph26):f3
```

```
  2u
  p56:gp6*1.0
  d16

  "DELTA = BigT - 2u - p56 - d16 + pwn*4.0/PI"
  DELTA
#endif /*gd_sel*/

/**************************************/
/*   TROSY-type N->H back transfer    */
/**************************************/
  (pwh ph3):f1

  /* shaped pulse */
  4u
  (pw_sl:sp13 ph4):f1
  2u pl1:f1
  /* end shaped pulse */

  2u
  p57:gp7
  d16

  "DELTA = taua - 4u - pw_sl - 2u - 2u - p57 - d16"
  DELTA

  (center (pwh*2 ph26):f1 (pwn*2 ph5):f3)

  2u
  p57:gp7
  d16

  DELTA

  /* shaped pulse */
  4u
  (pw_sl:sp13 ph26):f1
  2u pl1:f1
  /* end shaped pulse */

  (pwh ph26):f1
  (pwn ph6):f3

  2u
  p58:gp8
  d16

#ifdef gd_sel
  "DELTA = taua - 2u - p58 - d16"
  DELTA
#else
  "DELTA = taua - 2u - p58 - d16 - 4u - pw_sl1 - 2u"
  DELTA

  /* shaped pulse */
  4u
  (pw_sl1:sp14 ph7:r):f1
  2u pl1:f1
  /* end shaped pulse */
#endif /*gd_sel*/

  (center (pwh*2 ph26):f1 (pwn*2 ph26):f3)

#ifdef gd_sel
  "DELTA = taua - 2u - p58 - d16"
#else
  /* shaped pulse */
  4u
  (pw_sl:sp13 ph8:r):f1
```

```
  2u pl1:f1
  /* end shaped pulse */

  "DELTA = taua - 4u - pw_sl - 2u - 2u - p58 - d16 + pwh*2.0/PI - 4u - de"
#endif /*gd_sel*/

  2u
  p58:gp8
  d16

  DELTA

#ifdef gd_sel
  (pwn ph28):f3

  BigT1

  (pwh*2 ph26):f1

  "DELTA = BigT1 - 2u - p59 - d16 - pwh*2.0/PI - 4u - de"
  DELTA

  2u
  p59:gp9*EA
  d16

  4u BLKGRAD
#else
  4u BLKGRAD
  (pwn ph28):f3
#endif /*gd_sel*/

/****************************************/
/*   Test how much water left           */
/*   set d13=AQ and zgoptn -Dwater      */
/*   also use small value of rg         */
/****************************************/
#ifdef water
  if "nsdone == 0" {
    d13
    d1
    (pwh*0.1 ph26):f1
  }
  else {
    d13
  }
#endif

/**************************************/
/*   Signal detection and looping    */
/**************************************/
  go=2 ph31
  d11 mc #0 to 2

#if defined(cal_HB1) || defined(cal_HR1)
  F2QF(calclc(l2,1))
#else
  F2QF(calclist(H_offset,1))
#endif /*cal_HB1, cal_HR1*/
  F1EA(calgrad(EA) & calph(ph3, +180) & calph(ph4, +180) & calph(ph6, +180) & calph(ph
31, +180) & calclc(Nphase, 1), caldel(d0, +in0) & calph(ph1, +180) & calph(ph2, +180)
& calph(ph11, +180) & calph(ph12, +180) & calph(ph31, +180))

HaltAcqu, 1m
exit

ph1=1 3 2 0
ph2=1 3 0 2
ph3=1
```

```
ph4=3
ph5=0 0 0 0 1 1 1 1 2 2 2 2 3 3 3 3
ph6=3
ph7=2
ph8=2
ph11=1 1 2 2
ph12=1 1 0 0
ph21=2 1
ph22=3 1
ph26=0
ph27=1
ph28=2
ph29=3
ph31=3 1 2 0 1 3 0 2

;d1 : Repetition delay d1
;d3 : taua ~1/4J(NH) 2.70 ms
;d11 : disk read/write delay
;d14 : BigT1 set to 500 us
;d15 : BigT set to 850 us
;d16 : gradient recovery delay 200 us
;d17 : time_T1, time for CEST, typically several 100s of ms
;pl1 : tpwr - power level for 1H pulses
;pl3 : dhpwr2 -  power level for N hard pulses
;pl10 : tsatpwr - power level for water presat
;pl11 : tpwrmess - power level for Messerle purge
;spw13 : tpwrsl - power level for water selective pulse pw_sl
;spw14 : tpwrsl1 - power level for water selective pulse pw_sl1
;p1 : pwh - 1H 90 degree pulse
;p3 : pwn - 15N 90 degree pulse
;p13 : pw_sl
;p14 : pw_sl1
;p22 : pwc_ad -  adiabatic C180 pulse width
;p32 : pwn_sl for selective pulse on amide N15s
;p51 : gradient pulse 51                                [400 usec]
;p52 : gradient pulse 52                                [1000 usec]
;p54 : gradient pulse 54                                [256 usec]
;p55 : gradient pulse 55                                [1500 usec]
;p56 : gradient pulse 56                                [625 usec]
;p57 : gradient pulse 57                                [256 usec]
;p58 : gradient pulse 58                                [256 usec]
;p59 : gradient pulse 59                                [256 usec]
;spnam13 : shape for pw_sl
;spnam14 : shape for pw_sl1
;spnam22 : shape for pwc_ad
;spnam32 : shape for pwn_sl
;phcor7 : phase correction for ph7 of pw_sl1
;phcor8 : phase correction for ph8 of pw_sl
;cnst1 : offset (Hz) for B1 calibration from o1
;cnst16 : weak CEST 1H B1 field in Hz
;l2 : pointer to delay value for calibration of B1
;l3 : loop counter set internally - ignore
;l4 : number of pairs of 15N 180o pulses to apply for suppressing cross correlation
;infl : 1/SW(X) = 2*DW(X)
;in0 : 1/(2*SW(x))=DW(X)
;nd0 : 2
;ns : 2*n
;FnMODE : echo-antiecho

;use gradient ratio:    gp 6 : gp 9
;                         80 : -39.6

;for z-only gradients:
;gpz1: -25%
;gpz2: 15%
;gpz4: 23%
;gpz5: -70%
;gpz6: 80%
;gpz7: 60%
```

```
;gpz8: 15%
;gpz9: -39.6%

;use gradient files:
;gpnam1: SMSQ10.32
;gpnam2: SMSQ10.32
;gpnam4: SMSQ10.32
;gpnam5: SMSQ10.32
;gpnam6: SMSQ10.32
;gpnam7: SMSQ10.32
;gpnam8: SMSQ10.32
;gpnam9: SMSQ10.32

;zgoptns: Dfsat, Dfscuba, Dmess_flg, Df1180, Dgd_sel, Dc_flg, DN_sel, Dcal_HB1, DHstea
dy_flg, Dcal_HR1, Deta_flg
```

```
/* 13CH3_1H_CEST_lek_800_v2_cp

  This pulse sequence will allow one to perform the following experiment:

  2D 1H/13C to measure exchange via 13CH3 CEST

  Assumes that sample is specifically 13C labeled

    1H: O1 on methyl 1H resonances (around 0.8 ppm)
        pwh = p1 1H pw90 @  power level pl1 highest power (13W)

    13C: O2 centre at 20 ppm
        pwc = p2 13C pw90 @ power level pl2 highest power (175W)

  To callibrate B1 field: set -Dcal_HB1

  Modified based on 13CH3_1H_CEST_ty_800_cpz by TY on Aug 8th, 2017.
  This new version detects HzCz magnetization after CEST period,
  no longer require IPAP-type processing.

*/

;prosol relations=<triple>
;aqseq 321

#include <Avance.incl>
#include <Grad.incl>
#include <Delay.incl>

/***********************/
/*   Define phases     */
/***********************/
#define zero ph=0.0
#define one ph=90.0
#define two ph=180.0
#define three ph=270.0

/**************************/
/*    Define pulses       */
/**************************/
define pulse dly_pg1     /* Messrle purge pulse */
  "dly_pg1=5m"
define pulse dly_pg2     /* Messrle purge pulse */
  "dly_pg2=dly_pg1/1.62"
define pulse pwh
  "pwh=p1"               /* 1H hard pulse at power level pl1 (tpwr) */
define pulse pwc
  "pwc=p2"               /* 13C hard pulse at power pl2 (dhpwr) */

#if defined(water_flg) || defined(buffer_flg)
  define pulse pw_sl1
    "pw_sl1=p14"         /* eburp1 pulse, ~7000 us */
#endif

/**************************/
/*    Define delays       */
/**************************/
define delay hscuba
  "hscuba=30m"
define delay taua        /* 1 / 4J(XH)        */
  "taua=d3"              /* 1s/(4*125) = 2 ms */

#ifdef Met_flg
  "taua=taua*125/139"
#endif

#if !defined(cal_HB1) && !defined(cal_HR1)
  define delay time_T1
    "time_T1=d17"        /* CEST duration - typically 100-500 ms */
```

```
#endif

"d11=30m"
"in0=inf1/2"            /* t1/2 increment */
"TAU2=0.2u"

/*****************************************************************/
/* f1180: Start t1 at half dwell to get -90/180 phase correction */
/*        in F1 (15N) dim - set zgoptns -Df1180                  */
/*****************************************************************/
#if defined(f1180) && !defined(cal_HB1)
  "d0=in0/2"
#else
  "d0=0"
#endif

/**********************************/
/*    Define other parameters    */
/**********************************/
#ifdef cal_HB1
  define list<delay> time_cal=<t1list_calB1>
#else
#ifdef cal_HR1
  define list<delay> time_cal=<t1list_calR1>
#else
  define list<frequency> H_offset=<$FQ1LIST>
#endif /*cal_HR1*/
#endif /*cal_HB1*/

/***************************************************/
/* Assign cnsts to check validity of parameter range */
/***************************************************/
#ifdef fsat
  "cnst10=plw10"         /* tsatpwr - set max at 0.00005W */
#endif

#ifdef mess_flg
  "cnst11=plw11"         /* tpwrmess - set max at 1W */
#endif

#if defined(water_flg) || defined(buffer_flg)
  "cnst14=spw14"         /* tpwrsl1 - set max at 0.001W */
#endif

  "cnst21=plw21"         /* dpwr pl21 - set max at 4.5W */

/*************************************************/
/* Set the weak H-amide B1 field for irradiation */
/*************************************************/
"p15=1000000.0/(4.0*cnst16)"
"plw15=plw1*(p1)*(p1)/((p15)*(p15))"
"cnst15=plw15"

/***************************************/
/* Define macro for 3-9-19 WATERGATE */
/***************************************/
#define Watergate (center (pwh*0.231 ph26 d19*2 pwh*0.692 ph28 d19*2 pwh*1.462 ph26 d1
9*2 pwh*1.462 ph26 d19*2 pwh*0.692 ph28 d19*2 pwh*0.231 ph26):f1 (pwc*2 ph26):f2)

/****************************/
/* Initialize loop counters */
/****************************/
"l2=0"
"l3=1"

;"acqt0=0"
;baseopt_echo

1 ze
```

```
/**************************************/
/* Check validity of parameter range  */
/**************************************/
#ifdef fsat
  if "cnst10 > 0.001" {
    2u
    print "error: presat power pl10 is too large"
    goto HaltAcqu
  }
#endif

#ifdef mess_flg
  if "cnst11 > 1.0" {
    2u
    print "error: tpwrmess pl11 too large"
    goto HaltAcqu
  }
#endif

  if "dly_pg1 > 10m" {
    2u
    print "error:dly_pg too long !! "
    goto HaltAcqu
  }

#if defined(water_flg) || defined(buffer_flg)
  if "cnst14 > 0.01" {
    2u
    print "error: power level for eburp1 pulse is too large"
    goto HaltAcqu
  }
#endif

  if "cnst15 > 0.01" {
    2u
    print "error: power level for 1H cest is too large"
    goto HaltAcqu
  }

  if "cnst16 > 250" {
    2u
    print "error: cnst16 - weak 1H B1 field strength in Hz must be <= 250 Hz !!!  "
    goto HaltAcqu
  }

  if "cnst21 > 2.5" {
    2u
    print "error: dpwr pl21 too large"
    goto HaltAcqu
  }

  if "pwc > 20u" {
    2u
    print "error: pwc too large < 20 us"
    goto HaltAcqu
  }

#if !defined(cal_HB1) && !defined(cal_HR1)
  if "time_T1 > 0.5s" {
    2u
    print "error: time_T1 is too long"
    goto HaltAcqu
  }
#endif

  if "aq > 64m" {
    2u
    print "error: aq is too long"
    goto HaltAcqu
```

```
  }

  if "d1 < 1.0s" {
    2u
    print "error: d1 must be at least 1s !!!"
    goto HaltAcqu
  }

2 d11 do:f2
/*****************************************/
/* Continue to check real time variables */
/*****************************************/
#ifdef cal_HB1
  "DELTA = time_cal[l2]"
  if "DELTA > 500m" {
    2u
    print "error: time_cal (evolution time) during B1 calib is too long"
    goto HaltAcqu
  }
#endif

  20u fq=cnst1:f1        /* jump 1H carrier to water */

/******************/
/* Messerle purge */
/******************/
#ifdef mess_flg
  20u pl11:f1
  (dly_pg1 ph26):f1
  20u
  (dly_pg2 ph27):f1
  20u pl10:f1
#endif /*mess_flg*/

/*****************/
/* Presaturation */
/*****************/
#ifdef fsat
  4u pl10:f1
  d1 cw:f1 ph26
  4u do:f1
  2u pl1:f1
#ifdef fscuba
  hscuba
  (pwh ph27):f1
  (pwh*2 ph26):f1
  (pwh ph27):f1
  hscuba
#endif /*fscuba*/
#else
  2u pl1:f1
  d1
#endif /*fsat*/

  20u UNBLKGRAD

/***************************************/
/* Erase 13C equilibrium magnetization */
/***************************************/
  2u pl2:f2
  (pwc ph26):f2

  2u
  p50:gp0
  d16

/*************************************************/
/*  +/- phase cycle on 1H prior to CEST period    */
/*************************************************/
```

```
#ifdef Hsteady_flg
  20u fq=0:f1            /* jump 1H carrier back to methyl */

  (pwh ph26):f1
  (pwh*2 ph0):f1
  (pwh ph26):f1

  2u
  p50:gp0
  d16
#endif /*Hsteady_flg*/

/*****************************************/
/*   CEST portion on Hz magnetization   */
/*****************************************/
  2u pl15:f1
#ifdef cal_HB1
  "DELTA=time_cal[l2]"
  2u fq=cnst3:f1
  DELTA cw:f1 ph26
  2u do:f1
#else
#ifdef cal_HR1
  "DELTA=time_cal[l2]"
  DELTA
#else
  2u fq=H_offset:f1
  time_T1 cw:f1 ph26
  2u do:f1
#endif /*cal_HR1*/
#endif /*cal_HB1*/
  20u fq=cnst1:f1        /* jump 1H carrier to water */

  2u
  p50:gp0
  d16

/***********************************************************/
/*  Destroy water and buffer magnetization at the beginning  */
/***********************************************************/
#ifdef water_flg
  (pw_sl1:sp14 ph26):f1

  2u
  p50:gp0
  d16
#endif

#ifdef buffer_flg
  20u fq=cnst2:f1        /* jump 1H carrier to buffer position */
  (pw_sl1:sp14 ph26):f1

  2u
  p50:gp0
  d16
#endif

  20u fq=0:f1            /* jump 1H carrier back to methyl */
  2u pl1:f1 pl2:f2

/*****************************************/
/* Special setup for the reference plane */
/*****************************************/
#ifndef cal_HB1
#ifdef cal_HR1
  "l3=0"
#else
  if "abs(H_offset)*1e6 > 10000" {
    "l3=0"
```

```
  }
  else {
    "l3=1"
  }
#endif /*cal_HR1*/
#endif /*cal_HB1*/

  if "l3 == 0" {
    (pwh ph1):f1
    (pwh ph27 pwh*2 ph26 pwh ph27):f1

    2u
    p51:gp1
    d16

    "DELTA = taua - 2u - p51 - d16"
    DELTA

    (center (pwh ph27 pwh*2 ph26 pwh ph27):f1 (pwc*2 ph26):f2)

    DELTA

    2u
    p51:gp1
    d16
  }
  else {
    (pwh ph2):f1
  }
  0.2u

/********************/
/*   13C labeling    */
/********************/
#ifndef Hsteady_flg
  (pwc ph3):f2
#else
  (pwc ph4):f2
#endif

  "TAU1=larger(d0-pwh*2-pwc*2.0/PI,TAU2)"
  TAU1

  (pwh ph27):f1
  (pwh*2 ph26):f1
  (pwh ph27):f1

  TAU1

  (pwc ph5):f2
/********************************************************/
/*   C->H back transfer                                 */
/*   use -Dwtg_flg to achieve better water suppression  */
/********************************************************/
#ifdef wtg_flg
  2u
  p52:gp2
  d16

  "DELTA = taua - 2u - p52 - d16 - d19*3.367 + pwh*2.0/PI + 0.2u"
  DELTA

  Watergate

  2u
  p52:gp2
  d16

  "DELTA = taua - 2u - p52 - d16 - d19*3.367 - 4u - de - 2u"
```

```
  DELTA
#else
  2u
  p53:gp3
  d16

  "DELTA = taua - 2u - p53 - d16 + pwh*2.0/PI + 0.2u"
  DELTA

  (center (pwh ph27 pwh*2 ph26 pwh ph27):f1 (pwc*2 ph26):f2)

  2u
  p53:gp3
  d16

  "DELTA = taua - 2u - p53 - d16 - 4u - de - 2u"
  DELTA
#endif

  4u BLKGRAD
  2u pl21:f2

  go=2 ph31 cpds2:f2
  d11 do:f2 mc #0 to 2

#if defined(cal_HB1) || defined(cal_HR1)
  F2QF(calclc(l2,1))
#else
  F2QF(calclist(H_offset,1))
#endif
  F1PH(calph(ph3, +90) & calph(ph4, +90), caldel(d0, +in0) & calph(ph3, +180) & calph(
ph4, +180) & calph(ph31, +180))

HaltAcqu, 1m
exit

ph0=2 1
ph1=1 1 1 1 3 3 3 3
ph2=0 0 0 0 2 2 2 2
ph3=0 2
ph4=0
ph5=0 0 2 2
ph26=0
ph27=1
ph28=2
ph29=3
ph31=0 2 2 0 2 0 0 2

;d1 : repetition delay
;d3 : taua ~1/4J(CH) 2.00 ms
;d11 : delay for disk i/o, 30ms
;d16 : gradient recovery delay, 200us
;d17 : time_T1 <= 500 ms
;d19 : delay for binomial water suppression
;      d19 = (1/(4*|cnst1|)), cnst1 = distance between methyl and water (in Hz)
;pl1 : tpwr - power level for pwh
;pl2 : dhpwr - power level for hard 13C pulse pwc
;pl15 : power level for 1H cest
;pl21 : dpwr - power level for 13C decoupling cpd2
;spw14 : power level for eburp1 pulse
;spnam14: eburp1 pulse on water
;p1 : pwh
;p2 : pwc
;p14 : eburp1 pulse width, typically 7000u
;p15 : pw of 1H for cest element
;p21 : pw of 1H for cest element
;p50: gradient pulse 50                                [1000 usec]
;p51: gradient pulse 51                                [500 usec]
;p52: gradient pulse 52                                [800 usec]
```

```
;p53: gradient pulse 53                                  [500 usec]
;cpd2 : 13C decoupling according to program defined by cpdprg2
;pcpd2: 13C 90 degree pulse at pl21 for cpd2
;cnst1 : water(Hz) - methyl(Hz)
;cnst2 : buffer(Hz) - methyl(Hz)
;cnst3 : offset (Hz) for B1 calibration from o1
;cnst14 : power level in w for 1H selective pulse
;cnst15 : power level in w for weak 1H B1 cest field
;cnst16 : 1H B1 field to use for CEST, typically 15-50 Hz
;l2 : pointer to the delay value for calibration of B1
;l3 : loop counter set internally - ignore
;inf1 : 1/SW(X) = 2*DW(X)
;in0 : 1/(2*SW(x))=DW(X)
;nd0 : 2
;ns : 2*n
;FnMODE : States

;for z-only gradients:
;gpz0: 50%
;gpz1: 30%
;gpz2: 80%
;gpz3: 40%

;use gradient files:
;gpnam0: SMSQ10.32
;gpnam1: SMSQ10.32
;gpnam2: SMSQ10.32
;gpnam3: SMSQ10.32

;zgoptns : Df1180, Dfsat, Dmess_flg, Dfscuba, Dwater_flg, Dbuffer_flg, Dcal_HB1, Dwtg_
flg, DHsteady_flg, Dcal_HR1, DMet_flg
```