CONSERT Stop and start

Procedure description

CHANGE RECORDS

ISSUE	DATE	EVOLUTION	AUTHOR
V0.1	23/10/2013	DRAFT for internal validation	Y. Rogez
V0.2	25/10/2013	Review Alain Hérique	Y. Rogez
V1.0	29/10/2013	Reviewed final version	Y. Rogez

Table of Contents

This document describes the CONSERT Stop and Start commanding mode.

The Stop and Start procedure is a special way to command the CONSERT instrument. This is defined for particular use during SDL and FSS operation phases.

It allows to stop OCN and LCN radar signal transmission (CONSERT Emitting mode) with both orbiter (OCN) and lander (LCN) instruments clocks remaining synchronized, and to restart radar sounding at the commanded date.

Applicable Documents

[AD 1] RO-OCN-TN-3825 CONSERT User Manual

[AD 2] CONSERT FOP

Acronyms

CONSERT	Comet Nucleus Sounding Experiment by Radiowave Transmission
FSS	First Science Sequence phase
LCN	CONSERT Lander instrument
OCN	CONSERT Orbiter instrument
SDL	Separation Descent and Landing phase
TDW	Touch Down Window

List of figures

List of tables

1 CONSERT NEW COMMANDING MODE NEED

The CONSERT instrument has to operate during both SDL and FSS phases. Between these two operational phases is the Touch Down Window. During TDW, the lander's touch down occurs at an unknown and unpredictable time. CONSERT instrument must not transmit any signal during this critical period, in order to avoid any interference with lander S/S and some P/L instruments.

In a first approach, we could satisfy that constraints by turning OFF the CONSERT instrument just before TDW, and switch it back to ON just after the TDW:

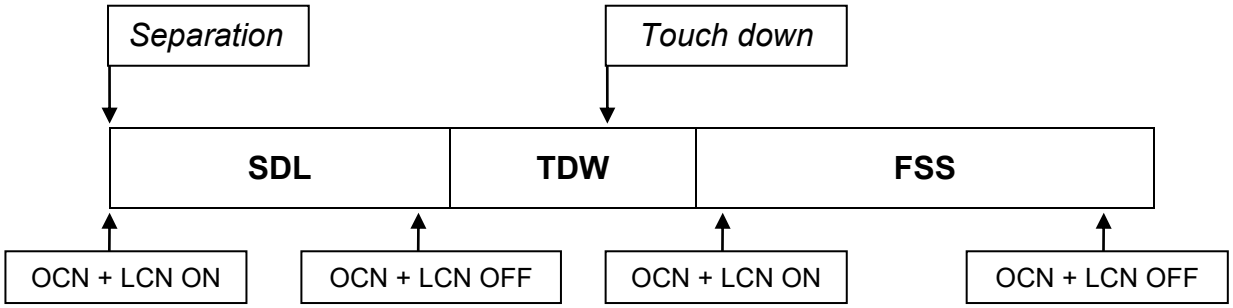

Figure 1: SDL/FSS with nominal commanding mode

This way of commanding present a major issue: when CONSERT instrument is switched ON, it needs a roughly 7 minutes window of lander/orbiter 20° angle visibility to proceed to its tuning phase. This performs the mandatory synchronization between OCN and LCN clocks.

These conditions are too constraining, even perhaps impossible, regarding to the global operation planning and to the unknown orbitography.

In order to ensure a correct functioning of CONSERT instruments for both SDL and FSS phases, we propose to put CONSERT instrument in a wait mode. This mode has the following specifications:

- No radar signal is transmitted.
- The transition from sounding mode to wait mode is defined by the TC reception time.
- The date of the transition from wait mode to sounding mode is defined by TC parameter. We have to specify the end time as a relative date by regards to CONSERT Turn-ON (and not a duration), in order to be sure that soundings restart after the end of TDW.
- It should minimize data and power budgets.

2 STOP AND START COMMANDING MODE FUNCTIONING SUMMARY

To achieve this, we define a new commanding mode, the so called “Stop and Start” for CONSERT.

In sounding nominal mode, the CONSERT instrument executes the following global process:

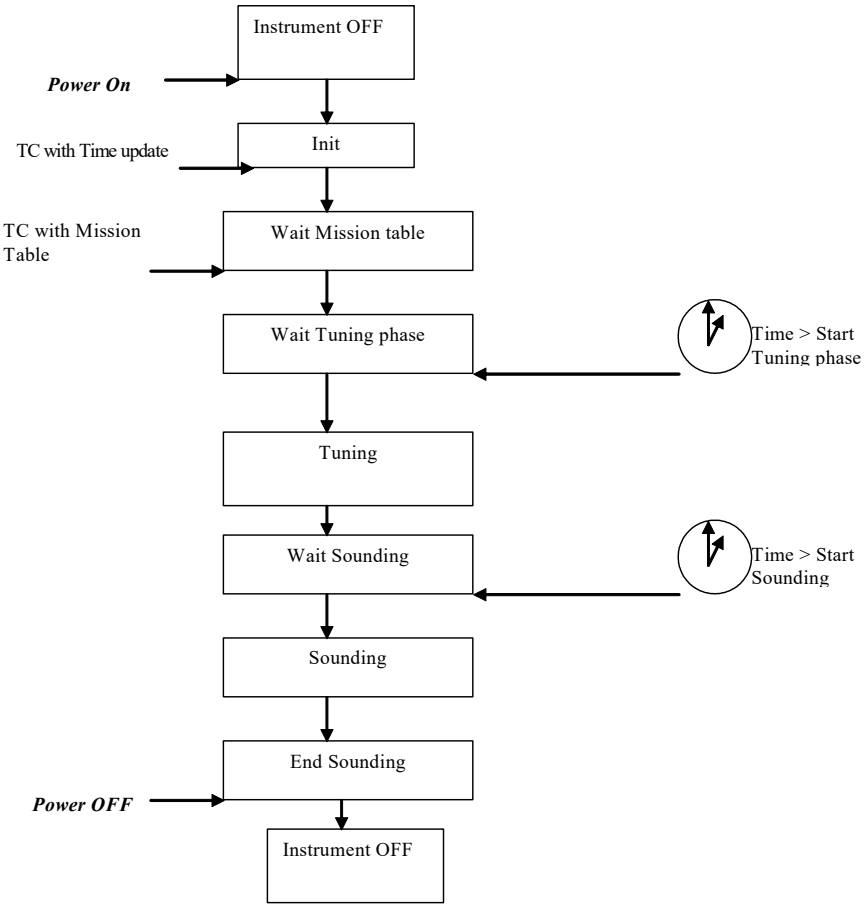

Figure 2: Phase Transition table

In the “Sounding” phase, CONSERT instrument performs an internal loop until the last sounding has been done. Each step of this loop executes the following:

1. Wait for the next sounding date (receiving TC if any)
2. Do the radar sounding
3. Increments the sounding number
4. Compute the **next sounding date** (expressed in CONSERT time base clock TIC)

1. Wait for the next sounding date
2. Do the radar sounding.
3. …

The idea is to make one of this loop’s steps lasts for a long period of time. Indeed, during (1), no sounding is done: the instrument works as if it is in wait mode. Only the clocks are going on, both on OCN and LCN, keeping their synchronization.

To be in wait mode until a specific time, we patch the value of **next sounding date** variable: we replace the computed "**next sounding date**" by a date that occurs just after the TDW.

The same restart date has to be patched on both OCN and LCN instruments parts.

The patched date is computed in CONSERT time base TIC relatively to the end of the Tuning (OCN and LCN internal time reset occurring 7 minutes after the CONSERT Turn-ON). The loop steps become the following:

1. Wait for the next sounding date: instrument processes the received patch TC and enlarge the step 1 up to the specified date in TC parameter.
2. Do the radar sounding
3. Increments the sounding number
4. Compute the **next sounding date** (expressed in CONSERT time base clock TIC)

This way of commanding the instrument is applicable to other operational phases, if needed.

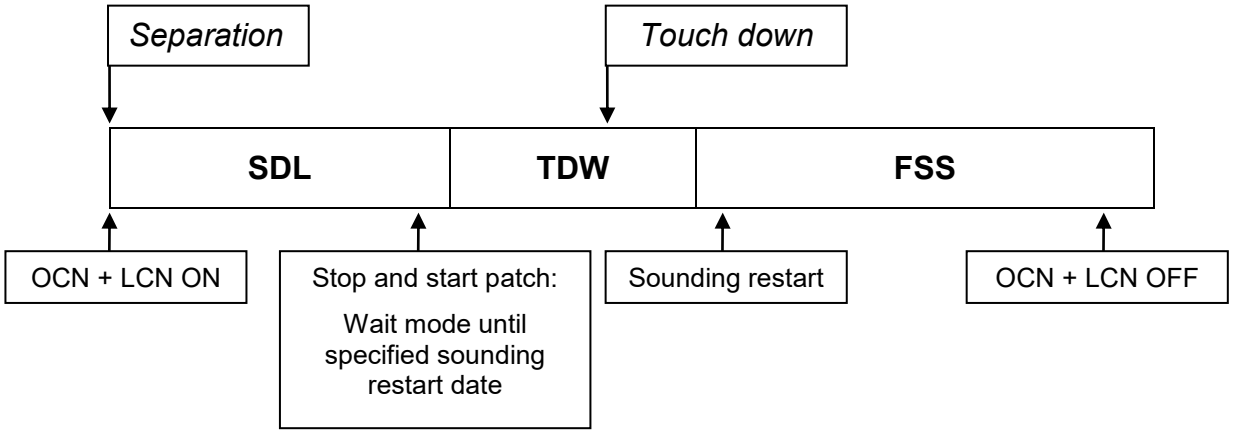

Figure 3: SDL/FSS with stop and start commanding mode

3 DETAILED DESCRIPTION

3.1 Nominal commanding mode

Relative Lander time	CONSERT time (TIC)	TC OCN & LCN	Instrument state	Remarks
0:00:00	0	**CONSERT ON**		Initialization and mission table update
			Waiting	
0:01:00	TUNETIC			The instrument is waiting for the date TIC = TUNETIC
			Tunning	
0:07:00	0			At end of tunning, both OCN and LCN clocks are synchronized
			Waiting	
0:10:00	STARTTIC			The instrument is waiting until the date (and not for a duration value) **NEXTTIC = STARTTIC** *STARTTIC is a TC parameter* At this time the sounding loop begins, defining the next sounding date to STARTTIC
			Waiting	
0:10:05	STARTTIC + DELTATIC		Sounding	**SOUNDINGNUMBER is incremented** The intrument is waiting until the next sounding date (and not for a duration value) The sounding is done
			Waiting	The next sounding date is computed, **NEXTTIC = STARTTIC +** **SOUNDINGNUMBER * DELTATIC** *DELTATIC is a TC parameter* and so on for all soundings…
			Sounding	
				Looping over a series of { Increment sounding number; Wait for date NEXTTIC = STARTTIC + SOUNDINGNUMBER * DELTATIC Sounding; }, until SOUNDINGNUMBER = NBSOUNDING NBSOUNDING is a TC parameter
			Waiting	After all soundings have been done, the instrument waits for the CONSERT-OFF
6:00:00	STARTTIC + NBSOUNDING * DELTATIC	**CONSERT OFF**		

Figure 4: Detailed nominal commanding mode

3.2 Stop and start commanding mode

Operation phase	Relative Lander time	CONSERT time (TIC)	TC OCN & LCN	Instrument state	Remarks
Separation	0:00:00	0	**CONSERT ON**	Waiting	
Descent	0:01:00	TUNETIC		Tunning	
	0:07:00	0		Waiting	
	0:10:00	STARTTIC		Waiting	
	0:10:05	STARTTIC + DELTATIC		Sounding	
				Waiting	
				Sounding	
		STARTTIC + N * DELTATIC *(N is the current SOUNDINGNUMBER)*			
TDW	1:30:00	STARTTIC + N_{stop} * DELTATIC *(N_{stop} is the SOUNDINGNUMBER when patch is applied)*	**CONSERT S&G PATCH**		During the waiting state of a sounding step, the buffered Stop&Start Patch TC is processed. We have DELTATIC duration to perform the patch insertion in a step. The patch will overwrite the NEXTTIC value in CONSERT program memory : Before the patch **NEXTTIC = STARTTIC + SOUNDINGNUMBER * DELTATIC** After the patch **NEXTTIC = PATCHVALUETIC** *PATCHVALUETIC is a TC parameter*
					During this waiting time, no sounding is done, only the instruments' clocks work, keeping the synchronization between OCN and LCN
FSS	1:45:00	PATCHVALUETIC		Waiting	The next sounding is done, with the delay specified by PATCHVALUETIC, and the nominal loop go on…
		PATCHVALUETIC + (N-N_{stop}) DELTATIC		Sounding	
					Note : the NUMBEROFSOUNDING value will be lower here than in the nominal case to achieve the same operating total duration, regarding to the PATCHVALUETIC duration.
	6:00:00	PATCHVALUETIC + (NBSOUNDING-N_{stop}) * DELTATIC	**CONSERT OFF**	Waiting	After all soundings have been done, the instrument waits for the CONSERT-OFF TC

Figure 5: Detailed start and stop commanding mode

4 ANALYSIS OF THE SOLUTION

4.1 Commanding method

- The solution to switch OFF CONSERT just before TDW to avoid interferences needs to restart and tune again at the beginning of the FSS, seems be impossible. The stop and start commanding mode allows achieving this objective and is considered as the baseline for SDL/FSS operations.
- The patch is performed with a relative date parameter, in the CONSERT time base TIC. Thus, we can guarantee:
  - the instant when radar sounding restart,
  - OCN and LCN are still synchronized for FSS.
- A single patch TC is needed, with a single parameter (one TC per electronics box)

4.2 Impact on constraints

- The patch TC's don't require a strongly accurate execution time. This date can differ for OCN and LCN without any influence on the restart date (assuming an execution time before patched value…). In this configuration Sounding Numbering can differ a little bit for OCN and LCN and coherence will be rebuilt from the timing.
- The total duration of CONSERT sounding can’t exceed roughly 30 hours. This value is constrained by the OCN and LCN clocks stability.
  With this commanding method, the waiting time during landing window is included in these 30 hours. So, the total number of sounding is reduced.
  SDL/FSS scenarios assumed by operation centres up to now fulfil this constraint.

  The total number of sounding has to be carefully calculated from the actual sounding duration = complete operation slot minus waiting duration. Additional margins have to be added between end-of-sounding and CONSERT Turn-OFF in order to be sure to have turn-off in "end-of-sounding" phase.
- The value of DELTATIC is defined for both SDL and FSS CONSERT operations and has to be set taking into account both SDL and FSS constraints. The SDL constraints concern interferences with SESAME calibration while FSS constraints concern spatial sampling of the comet nucleus for CONSERT tomography.

4.3 Budget

- Power budget during wait mode: ~3 W
- No telemetry produced during wait mode.

5 STOP AND START TELECOMMANDS

Patch orbiter

```
The FOP is under discussion with FCT
     00:00:00   ZCN00602                  \       # Load_Memory
                PCNG0600 = 3C             \       # Memory_block
                PCNG0601 = 01             \       # N
                PCNG0610 = 0000 500B      \       # MemStart
                PCNG0620 = 2              \       # MemLength
                PCNG0630 = Value          \       # MemData (32 bits)
```

Patch lander:

```
To be defined in the FOP (we prefer a dedicated SEQ)…
TC direct
               PARAM1 = 0204 500B
               PARAM2 = Value
```

The **Value** specified for **PCNG0630** (OCN) and **PARAM2** (LCN) are set to the date of the next sounding after patch execution.

This value is expressed in TIC since the end of the tuning.

Annex 1. CONSERT PROGRAM CODE

This section provides the section of on-board software which the Stop and Start patch will affect. It defines the Wait and Sounding loop.

5.1 CONSERT lander software

5.1.1 *LCN code excerpt*

SWL15 - main_l.c – function main() – l. 157-187

```
/*****************************************************************************/
/**************        MODE        SOUNDING                  *****************/
/*****************************************************************************/


/* preparation of soundings */
nextTIC = startTIC;  /* next action is start of first sounding */
status = set_bit(status,STAT_BIT_SOUNDING); /* status bit SOUNDING activated */

for (sounding_number = 1;sounding_number < total_soundings+1;sounding_number ++) /* main sounding
loop */
    {
    readTIC(); /* update time */
  /* enough time left before nextTIC, TC actions active */
    while (TIC < (nextTIC - EPSILON))
        {
        tc_activities();
        readTIC(); /* update time */
        }

 /*  loop waiting for next sounding start, no other Actions */
    do  { /*  loop waiting for tuning start */
        readTIC();
        }  /* end loop waiting for tuning start */
    while (TIC < nextTIC);
    /* a sounding is starting now */
    sounding();
    nextTIC = nextTIC + deltaTIC;
    }  /* end of main sounding loop */

status = clear_bit(status,STAT_BIT_SOUNDING); /* status bit SOUNDING cleared */
status = set_bit(status,STAT_BIT_END); /* status bit SOUNDING cleared */
```

The patched variable is `nextTIC`.

5.1.2 *LCN global variables declaration:*

d glob l.h Lander

```
/*      CONSERT     EQM software      */
/*      file name : def_gl.h          */
/*      version : swl 14                */
/*      edition date : 29/08/99       */
/*      last update : 06/07/2000        */
/*                          added global var timeout */
/*      include file for definition of global variables */


extern unsigned long   tuneTIC;                /* time in TIC of the start of Tuning mode */
extern BYTE            OCXO_freq;           /* freqency of the OCXO */
extern unsigned long   startTIC;               /* time in TIC of the start of first Soundings */
extern unsigned int    deltaTIC;     /* time interval in TIC between each sounding */
extern unsigned long   nextTIC;          /* time in TIC of the start of next Soundings */
extern unsigned int    sounding_number;  /* number of present sounding */
extern unsigned int    total_soundings;    /* total number of sounding */
extern BYTE            mini_att;             /* minimal attenuation */
extern BYTE            max_att;          /* maximum attenuation */
extern unsigned long   TIC;                              /* prestent time in TIC */
extern unsigned long   soundingTIC;        /* start time of present sounding, time in TIC */
extern unsigned int    packet_num;           /* TM packet number */
extern  BYTE           HK[BLOCK_LENGTH]; /* HK packet for standart TM*/
extern BYTE            T[NB_DATA];      /* data from FPGA */
extern BYTE            full_I_b1[STEPS];   /* table of I channel data in 16 bit, MSB */
extern BYTE            full_I_b0[STEPS];   /* table of I channel data in 16 bit, lsb */
extern BYTE            full_Q_b1[STEPS];   /* table of Q channel data in 16 bit, MSB */
extern BYTE            full_Q_b0[STEPS];   /* table of Q channel data in 16 bit, lsb */
extern BYTE            framed_I[STEPS]; /* table of I channel data framed to 8 bit */
extern BYTE            framed_Q[STEPS]; /* table of Q channel data framed to 8 bit */
extern BYTE            corel_I[STEPS];  /* table of I channel correlation data from FPGA */
extern BYTE            corel_Q[STEPS];  /* table of Q channel correlation data from FPGA */
extern BYTE            framing;                /* framing byte from FPGA */
extern BYTE            GCW;                  /* present value of hardware attenuation byte GCW  */
extern BYTE            status;                   /* status byte of yhe instrument */
extern BYTE            TC[BLOCK_LENGTH]; /* last recieved TC message */
extern BYTE             nbtc;              /* number of significant BYTEs in TC ,  if no TCs recieed
*/
extern BYTE            err_code;        /* error code of last occured error, 0 if no error */
extern BYTE            err_count;       /* error count since start */
extern BYTE            FIOW_ratio;                /*  from Mission table , 0 if no FIOW*/
extern BYTE            FIOW_count;                /*  form 0 to FIOW_ratio */
extern BYTE            mode_byte;              /* define of TEST MODES */
extern unsigned int    cor_modulus[STEPS];            /* correlation modulus I^2 + Q^2 */
extern  BYTE           timeout;                  /*timeout indicator from FPGA routines */
```

5.1.3 *LCN memory mapping*

Variable name	Address
tuneTIC	5000
OCXO_freq	5004
startTIC	5005
deltaTIC	5009
nextTIC	500B
total_soundings	500F
sounding_number	5011
mini_att	5013
max_att	5014
TIC	5015
soundingTIC	5019
packet_num	501D
HK	501F
T	505F
full_I_b1	5858
full_I_b0	5957
full_Q_b1	5A56
full_Q_b0	5B55
framed_I	5C54
framed_Q	5D53
corel_I	5E52
corel_Q	5F51
Framing	6050
GCW	6051
Status	6052
TC	6053
Nbtc	6093
err_code	6094
err_count	6095
FIOW_ratio	6096
FIOW_count	6097
mode_byte	6098
cor_modulus	6099
Timeout	6297

Table 1 : LCN memory mapping

5.2 CONSERT orbiter software

5.2.1 *OCN code excerpt*

SWO8 - main_o.c – function main() – l. 163-195

```
/*****************************************************************************/
/**************        MODE        SOUNDING                  *********************/
/*****************************************************************************/


/* preparation of soundings */
nextTIC = startTIC;  /* next action is start of first sounding */
status = set_bit(status,STAT_BIT_SOUNDING); /* status bit SOUNDING activated */
send_pro_report(EID_SOUND_START);   /* send a progress report */

for (sounding_number = 1;sounding_number < total_soundings+1;sounding_number ++) /* main sounding
loop */
    {
    readTIC(); /* update time */
  /* enough time left before nextTIC, TC actions active */
    while (TIC < (nextTIC - EPSILON))
        {
        tc_activities();
        readTIC(); /* update time */
        }

 /*  loop waiting for next sounding start, no other Actions */
    do  { /*  loop waiting for tuning start */
        readTIC();
        }  /* end loop waiting for tuning start */
    while (TIC < nextTIC);
    /* a sounding is starting now */
    sound_o();
    nextTIC = nextTIC + deltaTIC;
    }  /* end of main sounding loop */

status = clear_bit(status,STAT_BIT_SOUNDING); /* status bit SOUNDING cleared */
status = set_bit(status,STAT_BIT_END); /* status bit SOUNDING cleared */
send_pro_report(EID_SOUND_COMPLETED);   /* send a progress report */
```

The patched variable is `nextTIC`.

5.2.2 *OCN global variables declaration:*

d_glob_o.h Orbiter

```
/*      CONSERT     EQM software      */
/*     ORBITER                         */
/*      file name : d_glob_o.h          */
/*      version : swo 7                */
/*      edition date : 21/01/2000       */
/*      last update : 05/07/2000       */
/*            OBT_update  global var added   */
/*      include file for definition of global variables */


/*  GLOBAL VARIABLES   equivalences   */

extern unsigned long   tuneTIC;              /* time in TIC of the start of Tuning mode */
extern BYTE            OCXO_freq;           /* freqency of the OCXO */
extern unsigned long   startTIC;            /* time in TIC of the start of first Soundings */
extern unsigned int    deltaTIC;            /* time interval in TIC between each sounding */
extern unsigned long   nextTIC;            /* time in TIC of the start of next Soundings */
extern unsigned int    sounding_number;     /* number of present sounding */
extern unsigned int    total_soundings;     /* total number of sounding */
extern BYTE            mini_att;            /* minimal attenuation */
extern BYTE            max_att;            /* maximum attenuation */
extern BYTE            tune_NBL_level;      /* level to be reached in tuning phase NBL */
extern BYTE             tune_NBL_zero;      /*flour level to detect carrier off in tuning phase NBL
*/
extern unsigned long   TIC;                /* prestent time in TIC */
extern  unsigned long  soundingTIC;        /* start time of present sounding, time in TIC */
extern unsigned long   SCET;               /*  SCET seconds in 32 bits */
extern BYTE            SUBSEC;              /* subseconds fist byte in SCET */
extern unsigned long   TIC_offset;         /* TIC offset at SCET ref reception */
extern  unsigned long  tuneTIC_offset;      /* addition negative TIC offset for tuning correction*/
extern unsigned long   SCET_offset;        /*  SCET seconds in 32 bits at reception */
extern BYTE            SUBSEC_offset;       /*SCET offset subseconds fist byte in SCET */
extern  BYTE           OBT_updated;          /* 0 if no LOBT update recieved, 1 if recived */
extern unsigned int    tm_seq_count[16];     /* TM packet sequence count in ieach packet type*/
extern BYTE            T[NB_DATA];           /* data from FPGA */
extern BYTE            full_I_b1[STEPS];    /* table of I channel data in 16 bit, MSB */
extern BYTE            full_I_b0[STEPS];    /* table of I channel data in 16 bit, lsb */
extern BYTE            full_Q_b1[STEPS];    /* table of Q channel data in 16 bit, MSB */
extern BYTE            full_Q_b0[STEPS];    /* table of Q channel data in 16 bit, lsb */
extern BYTE            framing;             /* framing byte from FPGA */
extern BYTE            GCW;                 /* present value of hardware attenuation byte GCW  */
extern BYTE            status;             /* status byte of yhe instrument */
extern BYTE            TC[255];             /* last recieved TC message */
extern BYTE            mode_byte;          /* define of TEST MODES */
extern BYTE           TC_inprocess;         /* flag for TC in process */
extern unsigned long  TC_timeout;           /* timeout for inplote TC */
extern unsigned int   TC_count;            /* count of current TC in packet*/
extern unsigned int   TC_pk_id;          /*  pk id of last recieved TC */
extern unsigned int    TC_seq_countrol;   /*  sequence control of last recieved TC */
extern unsigned int   TC_pk_length;      /*  pk length of last recieved TC */
extern BYTE           level_GCW;            /* level detected at end of GCW phase*/
extern BYTE           level_zero;            /* level detected at end of carrier */
extern BYTE           tuning_inter;           /* intercallille of tuning PLL phasec */
extern BYTE           timeout ;              /* set to 1 if time out during FPGA activities */
```

5.2.3 *OCN memory mapping*

Variable name	Address
tunetic	5000
ocxo_offset	5004
startic	5005
deltatic	5009
Nextic	500b
sounding number	500f
total soundings	5011
min att	5013
max att	5014
tunenl level	5015
tune nbl zero	5016
tic	5017
sounding tic	501b
scet	501f
subscet	5023
tic_offset	5024
scet_offset	5028
tunetic offset	502c
subscet_offset	5030
obt_update	5031
tm_seq_count (16 words)	5032
T (2041 bytes RAZ apres lecture)	5052
full_ib0	584b
full_ib1	594a
full_qb0	5a49
full_qb1	5b48
framing	5c47
gcw	5c48
mode_byte	5c4a
status	5c49
Tc (255 bytes)	5c4b
tc_inprocess	5d4a
tc_timeout	5d4b
tc_count	5d4f
tc_pk_id	5d51
tc_seq_countrol	5d53
tc_pk_length	5d55
level gcw	5d57
level zero	5d58
tuning inter	5d59
time out	5d5a

Table 2 : OCN memory mapping

Annex 2. STOP AND START COMMANDING MODE TESTS : PHC

This commanding mode will be tested on board during PHC executing the sequence defined in ***PHC – S&S V1-1*** and written hereafter:

Orbiter Activities: orbiter schedulling

```
ACNS400A (Switch OCN ON and Mission table update)
ACNS400B (Turn OCN OFF - no parameters)
```

Lander Activities: Absolute Time Tagged commands

```
ALNS199A (Switch LCN ON - no parameters)
ALNS199B (Mission Table upload)
ALNS199C (Dump - no parameters)
ALNS199D (Turn LCN OFF- no parameters)
```

Mission Table: Usual parameters

```
Time Step in TIC = 4.95 seconds
PCNGA040 = PLNGA040 = 0BCD <hex> = 3021 <dec>      Time Step in TIC
```

Total Number of sounding: sample values

```
PCNGA050 = 0147 <hex> = 327 <dec>
PLNGA050 = 012F <hex> = 303 <dec>
```

Default Values for the other orbiter parameters

```
VCNDA011 = def val: 01 <dec>          Mission Table Index
VCNDA012 = def val: 00 <dec>          Pad
VCNGA020 = def val: 038C60 <hex>      Start TIC for Tuning Phase
VCNGA030 = def val: 008F0D <hex>
VCNDA061 = def val: 80 <hex>          Clock Initial Frequency
VCNDA062 = def val: 00 <dec>          Mode Setting
VCNDA071 = def val: 00 <dec>          Minimum GCW Attenuation
VCNDA072 = def val: 1F <hex>          Maximum GCW Attenuation
VCNDA081 = def val: 95 <hex>          TAB_NBL_LEVEL
VCNDA082 = def val: 85 <hex>          TAB_NBL_ZERO
```

Default Values for the other lander parameters

```
VLNDA012 = def val: 1 <hex>           Mission Table Index
VLNGA020 = def val: 35A4F <hex>       Start TIC for Tuning Phase
VLNGA030 = def val: 008F0D <hex>
VLNDA061 = def val: 83 <hex>          Clock Initial Frequency
VLNDA091 = def val: 5 <hex>           FIOW_RATIO
VLNDA062 = def val: 0 <hex>           Mode Setting
VLNDA071 = def val: 0 <hex>           Minimum GCW Attenuation
VLNDA072 = def val: 1F <hex>          Maximum GCW Attenuation
VLNDA092 = def val: 0 <hex>           MT_SPARE
```

Patch orbiter

```
The FOP has to be discussed with FCT, since there is no existing SEQ to patch 2 words. (32 bits)…
     00:00:00   ZCN00602                    \       # Load_Memory
                PCNG0600 = 3C               \       # Memory_block
                PCNG0601 = 01               \       # N
                PCNG0610 = 0000 500B        \       # MemStart
                PCNG0620 = 2                \       # MemLength
                PCNG0630 = Value            \       # MemData (32 bits)
```

For this test,
PCNG0630 = 0005 9683
PCNG0630 = 0010 C389
PCNG0630 = 0037 E11D

Patch lander:

```
To be defined in the FOP (we prefer a dedicated SEQ)…
TC direct
                PARAM1 = 0204 500B
                PARAM2 = Value
```

For this test,
PARAM2 = 0005 9683
PARAM2 = 0010 C389
PARAM2 = 0037 E11D

- END OF DOCUMENT -