



UiO : **University of Oslo**

Bartolomeo mNLP commands

DEPARTMENT OF PHYSICS, UNIVERSITY OF OSLO

Updated by mNLP group at UiO, version 10 Oct. 2022

The last firmware software version is **2022-06-29** (firmware date).

This document is partly a copy of the BRIK-II ScinMon Interface Control Document (BRIK-II-ISIS-ICD-400). Only the Software Interfaces are described in this document. It has been modified to show the Software Interfaces on the Bartolomeo mNLP instrument.

Contents

1	Software Interfaces	5
1.1	Data Interfaces.....	5
1.1.1	Command Interface	5
1.1.2	Science Data Interface	5
1.2	Mode Definition	5
1.3	Mode Transition	6
1.4	TT&C Interface.....	7
1.4.1	Byte Order: Big Endian	7
1.4.2	Negative Values: Two's Complement Encoding	7
1.4.3	Basic Data Exchange Logic	7
1.4.4	System Type, Interface Version and Board Identifier (STID, IVID, BID)	8
1.4.5	Interface Version Change List.....	9
1.4.6	Command Code (CC) and Response Code (RC)	9
1.4.7	Response STAT Byte	9
1.4.8	mNLP board identification	10
2	Command Overview.....	11
2.1	0xAA (0xAB) – System Reset	11
2.2	0x02 (0x03) – No Operation.....	12
2.3	0x10 (0x11) – Start Boom Deployment	13
2.4	0x12 (0x13) – Upload new application image	17
2.5	0x30 (0x31) – Switch to Idle Mode	20
2.6	0x32 (0x33) – Switch to Deploy Mode.....	20
2.7	0x34 (0x35) – Switch to Science Mode	21
2.8	0x40 (0x41) – Get Housekeeping Information.....	22
2.9	0x62 (0x63) – Get Configuration Parameters.....	28
2.10	0x64 (0x65) – Set Configuration Parameters.....	30
2.11	0x66 (0x67) – Set PPS Timestamp	34
2.12	0xF0 (0xF1) – Request Debug Message	36
2.13	0xF6 (0xF7) – Request Calibration Update.....	38
2.14	0x57 (0x58) – Run Bias Sweep	40
2.15	0x45 (0x46) – Get Temperature.....	43
3	Science Data Interface	44
3.1	Science packet (1024 bytes)	44
3.2	Housekeeping packet (109 bytes).....	44
3.3	Clock packet (12 bytes).....	45
3.4	Bias sweep packets (1024 bytes).....	45
3.5	Science data volume	46

1 Software Interfaces

The software interfaces between the payload and the platform deal with the control bus interaction and payload data interface.

1.1 Data Interfaces

There are two data interfaces:

- Command interface (over I²C)
- Data interface (over RS422)

The nature of these interfaces are described in the following sections.

1.1.1 Command Interface

The command and control interface allows configuring and controlling the mNLP payload.

- Boom deployment
- Health check/status
- Load parameters, one by one or complete list
- Electron emitter control
- Probe biases control
- Science/acquisition mode

The I²C address is 0x49 for the mNLP *BUS3* boards and 0x36 for the mNLP *BUS4* boards.

1.1.2 Science Data Interface

When the mNLP payload is nominally active, it will be acquiring science data. The science data is transmitted over the RS422 data interface.

The default baud rate is 921600. It can be changed with the RS422_BAUD_RATE in the *Set Configuration Parameters* command.

The science data is packet based.

1.2 Mode Definition

The mNLP payload supports a set of modes that clearly communicate the operational state. These modes are described in Table 1.

Table 1: mNLP Mode Definitions

ID	Name	Description
----	------	-------------

	Off	unpowered
0	Bootloader	In this mode, a new application image can be uploaded to the mNLP. Bootloader mode will transition to startup mode in one of two ways; Failing to send a valid command within the time limit or by commanding the bootloader to transition to startup mode. The bootloader only implement a subset of the commands
1	Startup	This mode is used to initialize the mNLP software. At the end of the startup the external command interface can be enabled to allow acceptance of external commands.
2	Idle	In this mode, the payload waits for a command from the master. This mode allows loading of new parameters, configuring controllers and executing calibration functions. Housekeeping telemetry can be requested in this mode.
3	Deploy	Only in this mode the deployment command is accepted which will initiate boom deployment. The deploy command will be rejected in all other modes. This makes Deploy mode an additional security mechanism to minimize chances of an accidental boom deployment occurring. Housekeeping telemetry can be requested in this mode.
4	Science	Upon switching to science mode the mNLP payload will start science data acquisition, and as a result will generate data over the data interface. Housekeeping telemetry can be requested in this mode.

1.3 Mode Transition

The possible mode transitions are described in Figure 1-1. Note that the transition back to off can occur from any mode (e.g. sudden reset) and is implied.

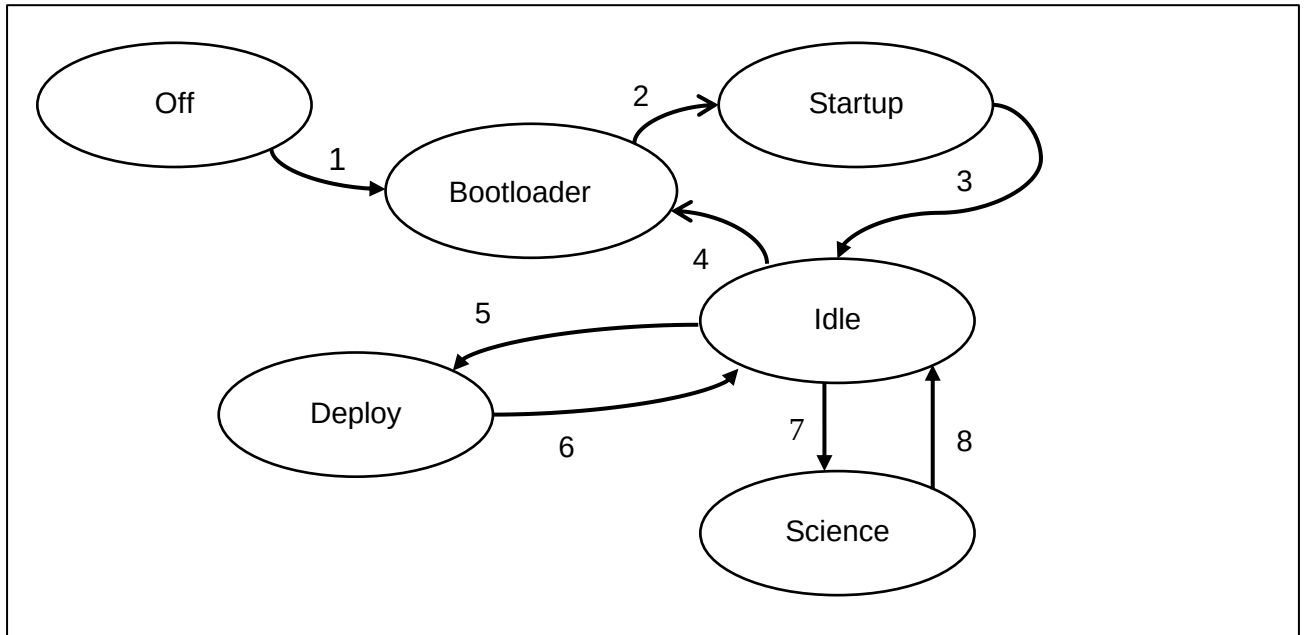


Figure 1-1: mNLP Mode Transition Diagram.

Table 1: mNLP Mode Transitions.

Transition	From	To	Trigger
1	Off	Bootloader	Power bus voltage exceeds power-on threshold
2	Bootloader	Startup	Master command, or failing to receive a valid master command within 6 seconds
3	Startup	Idle	Completion of startup
4	Idle	Bootloader	Master command. Transition time is 2 seconds
5	Idle	Deploy	Master command
6	Deploy	Idle	Master command, mode completion or error
7	Idle	Science	Master command
8	Science	Idle	Master command or mode completion

1.4 TT&C Interface

1.4.1 Byte Order: Big Endian

The byte order of multi-byte values is big endian. Hence, the most significant byte comes first and the least significant byte last.

E.g. value = 0xAABBCCDD, byte representation = [0xAA 0xBB 0xCC 0xDD]

1.4.2 Negative Values: Two's Complement Encoding

All negative values will be represented using two's complement encoding.

1.4.3 Basic Data Exchange Logic

A command is a data transfer from the master On Board Controller (OBC) to the mNLP payload, while a response is a data transfer from the mNLP payload to the master (OBC). Every command will produce a response. The flow is shown in Figure 1.2 .

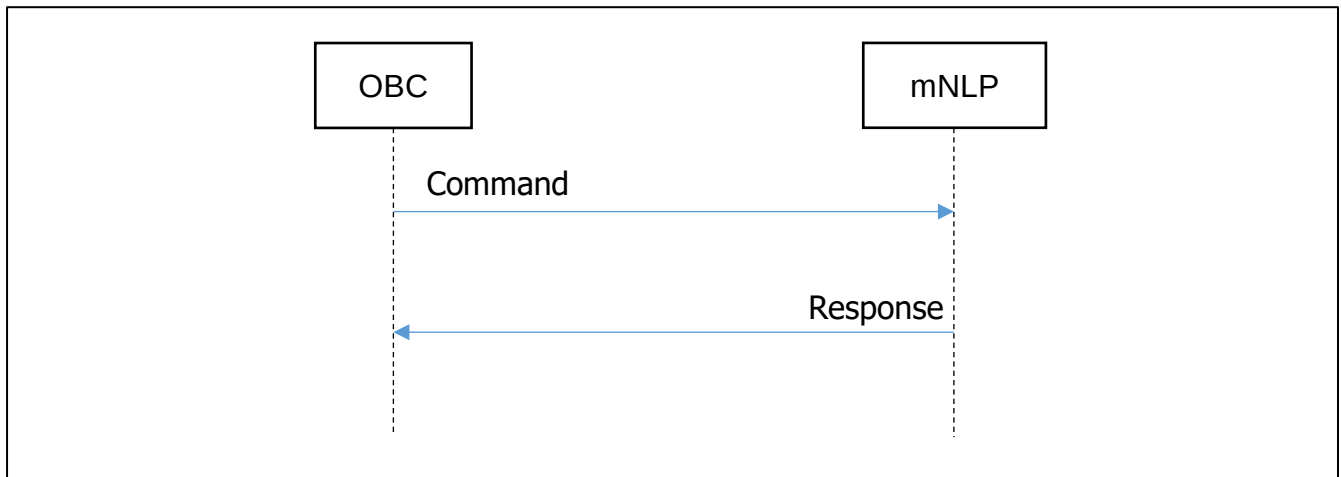


Figure 1-2 :Command-Response interaction.

The initial bytes of all commands have the same function, after which a variable amount of parameters can be defined depending on the need. The initial command bytes are : STID, IVID, BID, CC. The former three are described in Section 1.4 while the command code (CC) is described in Section 2.

Similarly, the initial bytes of all responses have the same function, after which a variable amount of parameters can be defined depending on the need. The initial response bytes are: STID, IVID, BID, RC, STAT. The former three are described in Section 1.4.4. The response code (RC) is described in Section 1.4.6. The STAT byte is described in Section 1.4.7.

1.4.4 System Type, Interface Version and Board Identifier (STID, IVID, BID)

The system type ID (STID) allows specification which subsystem the commands was intended for. When a command is received with an incorrect STID the command will be rejected. A STID of 0 is always accepted.

System type ID = 0xE2 (Multi Needle Langmuir Payload, mNLP)

The (software) interface version ID (IVID) specifies the interface version in use by the hardware. An IVID of 0 is always accepted.

Interface version ID = 0x01 (First interface version)

The board ID (BID) uniquely identifies a board of a given STID within the platform. When a command is received with an incorrect BID the command will be rejected. A BID of 0 is always accepted.

Board ID = 0x01 (mNLP board 1)

The Board ID is 0x01 for the mNLP *BUS3* boards and 0x02 for the mNLP *BUS4* boards.

1.4.5 Interface Version Change List

Changes to the software interface are tracked in the following table. The highest IVID shown reflects the version of the interface described in this document. Document copies (or portions of) with older interface versions should be kept for reference in case multiple version are 'in the wild' at the same time.

Table 0-2: Interface Version Change Log.

IVID	
0x01	First Version

1.4.6 Command Code (CC) and Response Code (RC)

The Command Code (CC) uniquely identifies a command for a subsystem type and interface version. Similarly a Response Code (RC) uniquely identifies a response for a subsystem type and interface version.

The following relation exists between a command and a response: $RC = CC + 1$.

1.4.7 Response STAT Byte

The status byte is always part of a response packet. Groups of bits within this byte have a specific meaning. The grouping is presented in Table 0-3, with the details discussed below.

Table 0-3: STAT bit field.

7	6	5	4	3	2	1	0
NEW	(reserved)			CMD ERROR #			

STAT NEW - Used with any command. NEW = 1 when the STAT value is retrieved for the first time after a command completes. Subsequent calls that get the STAT byte get NEW = 0.

STAT CMD ERROR # - The immediate result of the command indicating whether the command was accepted for processing. The available codes are shown in Table 0-4.

Table 0-4: STAT CMD Error Field.

CMD Error #	Description
0	accepted
1	rejected
2	rejected: invalid command code
3	rejected: parameter missing

4	rejected: parameter invalid
5	rejected: CC unavailable in current mode/configuration
6	rejected: invalid system type or interface version
7	internal error occurred during processing

1.4.8 mNLP board identification

The mNLP board has four *jumper*s. They are named:

ADDR_SEL_1
ADDR_SEL_2
BRD_ID_1
BRD_ID_2

The *ADDR_SEL* jumpers are used by the software to select the I²C address.
The *BRD_ID* jumpers are used by the software to select the Board ID.

The default I²C address is 0x49. When the *ADDR_SEL_1* is on, the software set the I²C address to 0x36.

The default Board ID is 0x01. When *BRD_ID_1* is on, the software change the Board ID to 0x02.

The mNLP BUS3 boards have (*Board ID 0x01, I²C address 0x49*)

ADDR_SEL_1 : off
ADDR_SEL_2 : off
BRD_ID_1 : off
BRD_ID_2 : off

The mNLP BUS4 boards have (*Board ID 0x02, I²C address 0x36*)

ADDR_SEL_1 : on
ADDR_SEL_2 : off
BRD_ID_1 : on
BRD_ID_2 : off

2 Command Overview

The available commands are tabulated in Table 2-1.

Table 2-1: Command Overview.

Command	Section	Comments
<i>Operational Commands</i>		
0xAA (0xAB) – System Reset	2.1	Start bootloader
0x02 (0x03) – No Operation	2.2	
0x10 (0x11) – Start Boom Deployment	2.3	
0x12 (0x13) – Upload new application image	2.4	Accepted by bootloader only
<i>Mode Switch Commands</i>		
0x30 (0x31) – Switch to Idle Mode	2.5	Accepted by bootloader also
0x32 (0x33) – Switch to Deploy Mode	2.6	
0x34 (0x35) – Switch to Science Mode	2.7	
<i>Data Request Commands</i>		
0x40 (0x41) – Get Housekeeping Information	2.8	
<i>Configuration Commands</i>		
0x62 (0x63) – Get Configuration Parameters	2.9	
0x64 (0x65) – Set Configuration Parameters	2.10	
0x66 (0x67) – Set PPS Timestamp	2.11	
0xF0 (0xF1) – Request Debug Message	2.12	
0xF6 (0xF7) – Request Calibration Update	2.13	Accepted by bootloader only
0x57 (0x58) – Run Bias Sweep	2.14	
0x45 (0x46) – Get Temperature	2.15	

2.1 0xAA (0xAB) – System Reset

Write length: 5 bytes.

Read length: 5 bytes.

Action: Perform a reset of the subsystem. System returns to bootloader mode within 2 seconds.

Note: A reply to this command will not always be retrievable, given that the system will shut down the command interface while preparing for the hard reset. After reset the reply will not be available anymore, instead having no response available (i.e. returns 0xFF on READ)

Command:

Name	Offset [Bytes]	Size [Bytes]	Type	Description
STID	0	1	uint8	System type ID.

IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header.
RSTKEY	4	1	uint8	Reset key. The value of this parameter needs to be set to 0xE2 for the command to be accepted. Any other value will cause the command to be rejected.

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0xAA 0xE2}

2.2 0x02 (0x03) – No Operation

Write length: 4 bytes.

Read length: 5 bytes.

Action: perform a NOOP command. Does not affect the operation of the subsystem. Used for quick checks on subsystem availability (i.e. a ping command).

Note:

A useful method to retrieve STID, IVID and BID of a subsystem is a NOOP command specifying the former as 0.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0x02}

2.3 0x10 (0x11) – Start Boom Deployment

Write length: 5 bytes.

Read length: 5 bytes.

Action: Start boom deployment. This command is only accepted when in deploy mode, and rejected otherwise. In order to release booms, the command must be sent twice within 2 seconds. In the first transmit the KEY must be set to ARM, in the second transmit the KEY must be set to RELEASE. Any other KEY values received will switch mode back to Idle.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
KEY	4	1	uint8	ARM = 0x33, RELEASE = <i>see table below</i>

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).

STAT	4	1	uint8	Command acceptance status and new indicator.
------	---	---	-------	--

RELEASE bit	Action
0 (LSB)	release cassette 1 primary
1	release cassette 1 secondary
2	release cassette 2 primary
3	release cassette 2 secondary
4	override release detect
5	1
6	0
7 (MSB)	1

The mNLP BUS4 instrument has one (1) boom cassette. The mNLP BUS3 instrument has two (1 and 2) boom cassettes. The main release is named primary (A), and the redundant release is named secondary (B). By setting the release bits the user can select what to release.

The 8-bit parameter BOOM_RELEASE_COUNT determines the maximum length of the release pulse in units of 5 [ms]. When BOOM_RELEASE_COUNT is set to 255 (maximum) the release pulse can be as long as 1.275 seconds. See section 2.10 for explanation of the parameters.

The default value of BOOM_RELEASE_COUNT is 0 .

override bit = 0:

When the *0x10 - Start Boom Deployment* RELEASE command is executed the release function (bartolomeo_antenna_release()) is executed. Then the detection of if the cassette already is released is done. The RELEASE status bits (b6 - b15) in the housekeeping are set. If allowed the release pulse is sent to the primary or secondary on the selected boom cassette.

The software checks if the boom cassette is already released. If it is already released, the deployment is not executed. 0 is then written to the housekeeping data.

If the cassette is not already released it sends a release pulse of length **5 [ms]/step * BOOM_RELEASE_COUNT step** milliseconds. If after each 5 [ms] step, release is detected, it set the release pulse low. The actual release time in [ms]/5 is written to the housekeeping data.

If after BOOM_RELEASE_COUNT steps the release is not detected, the software gives up, and set the release pulse low. 255 is then written to the housekeeping data.

(However, it takes some milliseconds for the software to detect the release. If the release was close to the end of the release pulse, the detection is done after the release pulse is finished and 255 (0xFF) is written to the housekeeping data.)

Examples (override bit is 0):

BOOM_RELEASE_COUNT = 12 in the *Set Configurations parameters* command sets the maximum release pulse to 60 milliseconds.

0xE2 0x01 0x01 0x10 0xA1 release cassette 1 primary

0xE2 0x01 0x01 0x10 0xA2 release cassette 1 secondary

0xE2 0x01 0x01 0x10 0xA4 release cassette 2 primary

0xE2 0x01 0x01 0x10 0xA8 release cassette 2 secondary

override bit = 1:

The software does **NOT** check if the boom cassette is already released. It sends a **BOOM_RELEASE_COUNT * 5** milliseconds release pulse. BOOM_RELEASE_COUNT is then written to the housekeeping data. If both primary and secondary bits are set, it will do this twice for the same boom cassette.

Examples (override bit is 1):

BOOM_RELEASE_COUNT = 20 in the *Set Configurations parameters* command sets the maximum release pulse to 100 milliseconds.

0xE2 0x01 0x01 0x10 0xB1 release cassette 1 primary

0xE2 0x01 0x01 0x10 0xB2 release cassette 1 secondary

0xE2 0x01 0x01 0x10 0xB4 release cassette 2 primary

0xE2 0x01 0x01 0x10 0xB8 release cassette 2 secondary

The time detected by software between ARM and RELEASE is written to the housekeeping data as [ms]/10 .

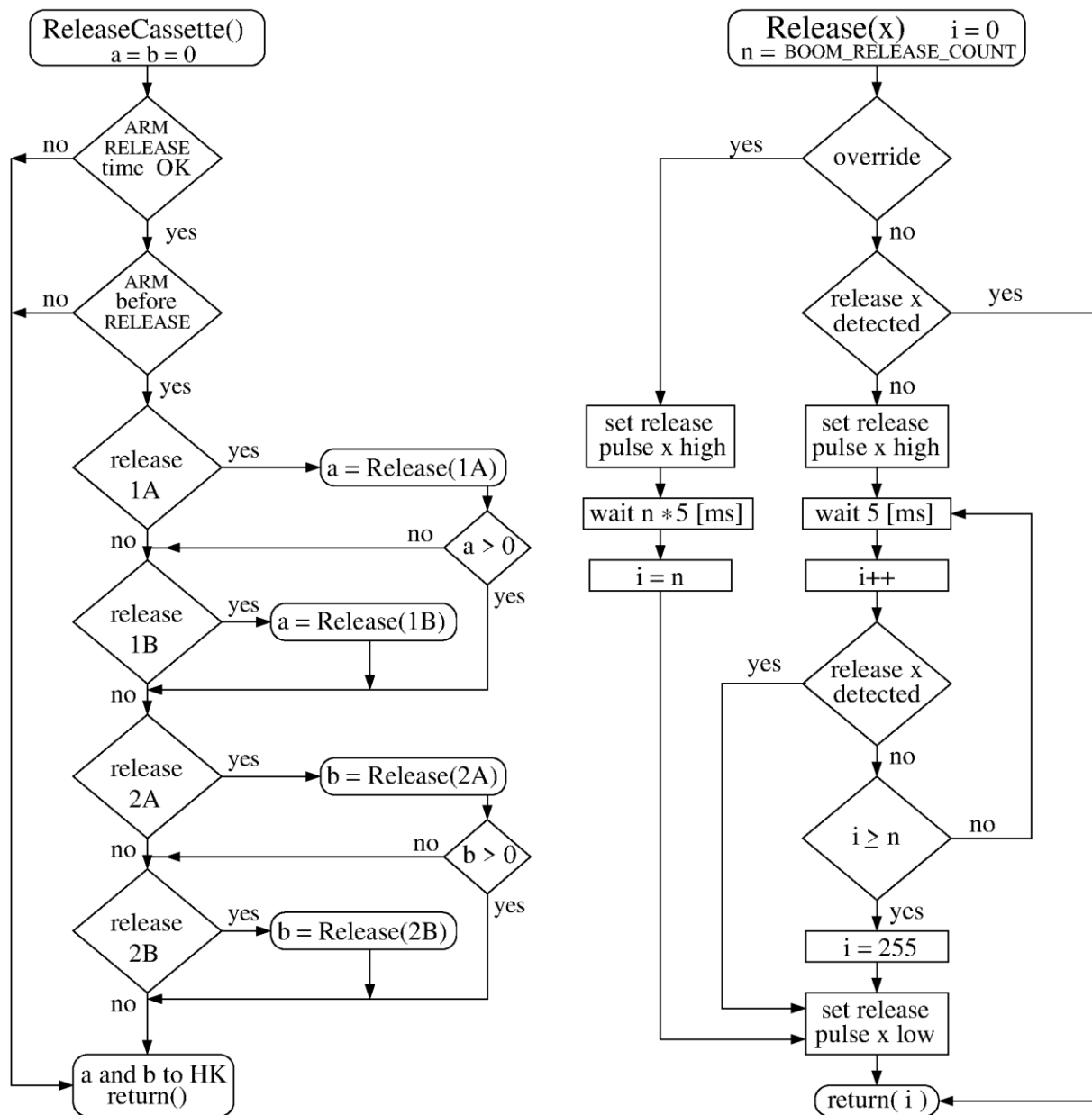


Figure 2.1: The boom release software logic flow diagram.

The variable *i* returned from the `Release(x)` function is written to the housekeeping `DEPLOY_1_TIME` and `DEPLOY_2_TIME`, for boom cassette 1 and 2 respectively. The time between the `ARM` and `RELEASE` command is written to the housekeeping `ARM_RELEASE_TIME`.

When we have not used the 'Start Boom Deployment' command since the mNLP instrument was booted, the `DEPLOY_1_TIME` and `DEPLOY_2_TIME` are 0 when the cassette 1 and/or cassette 2 are not released. The `DEPLOY_1_TIME` and `DEPLOY_2_TIME` are 1 when cassette 1 and/or cassette 2 are released

The housekeeping status byte is also set from the boom release software, see the table in Sect. 2.8.

(Note: The pinpuller, which release the booms, breaks the release pulse when it starts moving.)

Usage Example:

Transfer the following bytes to the system.

```
{<STID> <IVID> <BID> 0x33
<wait 400 [ms]>
<STID> <IVID> <BID> 0xA1}
```

2.4 0x12 (0x13) – Upload new application image

Write length: 68 bytes.

Read length: 5 bytes.

Action: Upload part of new application image to bootloader (Only in bootloader mode)

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
IMAGE_PART	4	64	uint8	Application image part where the packet SEQUENCE_NUMBER is equal to previous packet SEQUENCE_NUMBER plus exactly 1, and first SEQUENCE_NUMBER sent after a reboot is 0x0000

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0x12 <IMAGE_PART>}

Procedure for uploading a new application image to mNLP

The size of the application image is in the order of ~100k bytes. The application image length will always be a multiple of 64 bytes, which is the size of SEQUENCE_NUMBER + APPLICATION_SEQUENCE.

Application image structure:

<i>Name</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
SEQUENCE_NUMBER	2	uint16	Sequence number for the following 62 bytes of the APPLICATION_SEQUENCE
APPLICATION_SEQUENCE	62	uint8	Actual application image part
CRC	4	uint8	32 bit CRC of all APPLICATION_SEQUENCE excluding the SEQUENCE_NUMBER

The application consists of a repeated pattern of <SEQUENCE_NUMBER> followed by the <APPLICATION_SEQUENCE>, where the final four bytes of the final APPLICATION_SEQUENCE is the CRC used by the mNLP to validate the application image.

Example file:

```
00000000: 0000ea00 614aeaff fffeea00 1262eaff fffeea00 659eea00
00000018: 67d5e51f f1b0e51f f1b00001 aa7e5ba4 9bb40000 00000000
00000030: 0000e92d 4000e24d d024e58d 1014e58d 00010010 e3e0c000
00000048: e58dc01c e59dc010 e59cc014 e35c0000 1a000001 ef000005
00000060: eaffffff e59dc010 e59cc014 e35c0000 0a000023 e3a000a4
00000078: eb001389 e58d0018 0002e59d c018e35c 00000a00 001de59d
00000090: c010e59d 0018e59c c014e580 c084e59d c018e3a0 0001e5cc
```

.

.

```
0001b7b0: 5f636f6e 66696775 72650076 65726966 06df795f 766f6c74
0001b7c8: 61676573 00766572 6966795f 63757272 656e7473 0069736c
0001b7e0: 32323331 375f696e 69740064 61635f73 65745f62 69617300
0001b7f8: 73645f63 6172645f 06e07465 73740061 64373736 385f696e
0001b810: 69740062 6f617264 5f746173 6b00626f 6172645f 72756e00
0001b828: 6567756e 5f6f6e00 a5a5a5a5 a5a5a5a5 a5a5a5a5 46e6e1d0
```

SEQUENCE_NUMBER, APPLICATION_SEQUENCE, CRC

Bootloader will not accept out of order <SEQUENCE_NUMBER>

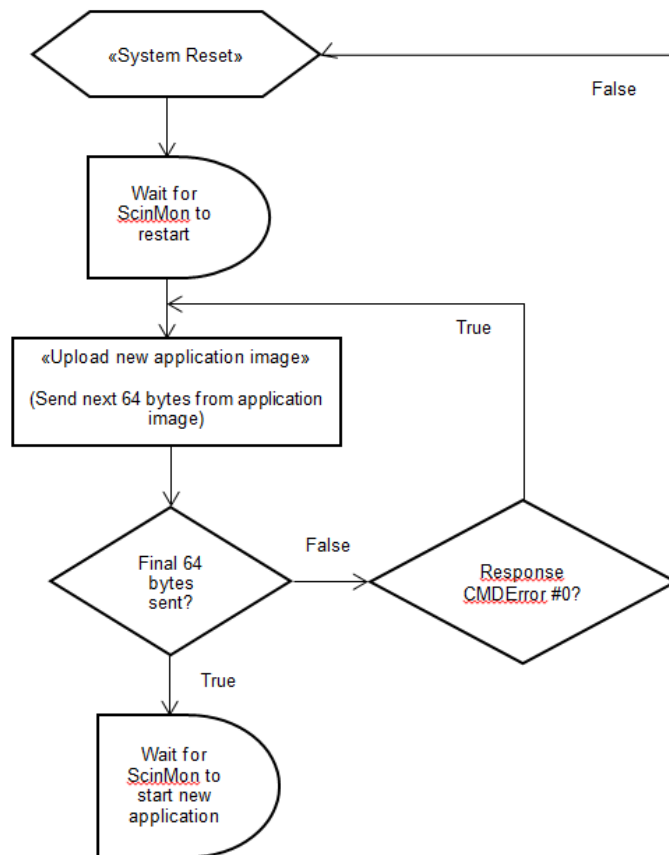


Figure 2.2 Upload new application

The mNLP should NOT be powered off during the upload process. A powerloss during upload will result in an aborted reflashing, and all received data are discarded. mNLP will on the next power up boot into bootloader and the upload process can be started from the beginning.

In the event of a powerloss in the short time of the actual memory flashing, which is right after reception of the last package, the old application might have been deleted without the new one being fully written. In this case the application is corrupted, but the bootloader is not impacted, and a new upload has to be performed.

Before uploading a new application program in space, the software has to be properly tested at the ground level.

Remember the bootloader use I²C slave address 0x49. If the board use a different I²C address, change to 0x49 immediately after the *System Reset* command is sent. Then upload the new application image to the mNLP instrument with the *Upload new application* command.

2.5 0x30 (0x31) – Switch to Idle Mode

Write length: 4 bytes.

Read length: 5 bytes.

Action: Cancel any action and return to Idle mode within 1 seconds.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0x30}

2.6 0x32 (0x33) – Switch to Deploy Mode

Write length: 4 bytes.

Read length: 5 bytes.

Action: Switch to Deploy mode. This mode, and only this mode, the boom deploy command will be accepted.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.

CC	3	1	uint8	See code in header
----	---	---	-------	--------------------

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0x32}

The software do not read voltages and currents between *0x32 - Switch to Deployment Mode* command and the two *0x10 - Start Boom Deployment* commands (ARM and RELEASE).

2.7 0x34 (0x35) – Switch to Science Mode

Write length: 6 bytes.

Read length: 5 bytes.

Action: Switch to science mode. The system will start acquisition of science data and produce data over the data interface. Exiting this mode will stop data to be transferred over the data interface.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
N_PACKETS	4	2	uint16	0: Continuous transfers of science data, no automatic return to Idle Mode. All other values are the number of packets to send before returning to Idle Mode

The default is the AD7768 Analog-to-Digital converter (ADC) is used. When

N_PACKETS is 1 the LTC2344 ADC is used. In this case it also continuous transfer science data. The science mode with the LTC2344 ADC has to be started before running the probe bias sweeps.

In science mode the mNLP instrument must have +5V and +/- 12V supply voltages.

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

Usage Example:

Transfer the following bytes to the system.

{<STID> <IVID> <BID> 0x34 0x00 0x00}

2.8 0x40 (0x41) – Get Housekeeping Information

Write length: 4 bytes.

Read length: 110 bytes.

Action: Retrieve system housekeeping information.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
-------------	---------------------------	-------------------------	-------------	--------------------

STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.
MODE	5	1	uint8	Return <i>operating mode</i> in the lowest 4 bit, bit 0 ... bit 3. See Sect. 1.2 for definition of <i>the operating mode</i> . See Sect. 3.2 for definition of the MODE bits.
1V2_VOLTAGE	6	2	uint16	1V2 rail voltage measured [mV]
3V3_CURRENT	8	2	uint16	3V3 rail current measured [mA]
12V_VOLTAGE	10	2	uint16	12V rail voltage measured [mV]
N12V_VOLTAGE	12	2	uint16	Negative 12V rail voltage measured [mV]
EGUN_CURRENT	14	2	uint16	Electron emitter current measured [mA]
5V_CURRENT	16	2	uint16	5V input current measured [mA]
FIL1_CURRENT	18	2	uint16	Electron emitter (EGUN) filament 1 current measured [mA]
FIL2_CURRENT	20	2	uint16	EGUN filament 2 current measured [mA]
1V2_CURRENT	22	2	uint16	1V2 rail current measured [mA]
5V_VOLTAGE	24	2	uint16	5V input voltage measured [mV]
IMAGE_SIZE	26	4	uint32	Application image size
IMAGE_TIMESTAMP	30	4	uint32	Application image creation timestamp
DEPLOY_1_TIME	34	1	uint8	Release time of boom cassette 1 [/5 ms] ¹⁾
DEPLOY_2_TIME	35	1	uint8	Release time of boom cassette 2 [/5 ms] ¹⁾
ARM_RELEASE_TIME	36	1	uint8	Time between ARM and RELEASE [/10 ms]
TBD	37	1	uint8	Not in use
TEMP1_THERMISTOR	38	2	uint16	Thermistor 1 temperature [°C] (encoded)

TEMP2_THERMISTOR	40	2	uint16	Thermistor 2 temperature [°C]
BIAS1_VOLTAGE	42	2	uint16	Bias voltage probe 1 [mV] (encoded)
BIAS2_VOLTAGE	44	2	uint16	Bias voltage probe 2 [mV]
BIAS3_VOLTAGE	46	2	uint16	Bias voltage probe 3 [mV]
BIAS4_VOLTAGE	48	2	uint16	Bias voltage probe 4 [mV]
GAIN	50	1	uint8	Gain in pre-amplifiers
SAMPLING_RATE	51	1	uint8	AD7768 ADC sampling rate (lookup table index, <i>see Sect 2.10</i>)
UPTIME	52	4	uint32	Time in milliseconds since mNLP transitioned from Startup to Idle
PPS_TIMESTAMP	56	4	uint32	PPS timestamp received
UPTIME_PPS_TIMESTAMP	60	4	uint32	UPTIME when PPS timestamp was received
BOARD_HIGH_MARK	64	2	uint16	board high mark
SCIENCE_HIGH_MARK	66	2	uint16	science high mark
3V3_VOLTAGE	68	2	uint16	3V3 rail voltage measured [mV]
12V_CURRENT	70	2	uint16	12V rail current measured [mA]
N12V_CURRENT	72	2	uint16	Negative 12V rail current measured [mA]
2V5_REF_VOLTAGE	74	2	uint16	measured 2V5 ref. (2.5 Volt) directly from the 12bit ADC, 0 is 0.0 V and 4095 is 5.0 V
LTC2344_SAMPLE_FREQ	76	2	uint16	The LTC2344 ADC sample frequency in Hz.
BOARD_STATUS	78	4	uint32	Board status bits b0 – b31. 0 denote status bit OK, 1 denote warning.
FIL1_VOLTAGE	82	2	uint16	EGUN filament 1 voltage measured [mV]
FIL2_VOLTAGE	84	2	uint16	EGUN filament 2 voltage measured [mV]
SWEEP_MODE	86	2	uint16	Sweep mode, <i>see Sect. 2.14</i> Run Bias Sweep
SAMPLE_FREQ	88	2	uint16	Sweep sample frequency in Hz
STEP_LENGTH_TIME	90	2	uint16	Sweep step length in unit of 0.1 us (fast sweep) or 0.01792 ms (slow sweep)
SAMPLE_HOLD_TIME	92	2	uint16	Sweep sample hold time in unit of 0.1 us (fast sweep) or 0.01792 ms (slow sweep)
SWEEP_NO	94	4	uint32	Number of sweeps

BIAS_START_VOLTAGE	98	2	uint16	Bias probe voltage start in sweeps
MAJOR_STEP_SIZE	100	2	uint16	Major step bias voltage size in sweeps
MAJOR_STEP_NO	102	2	uint16	Number of major bias steps in sweeps
MINOR_SWEEP_START	104	2	uint16	The minor sweep starts after # major steps in sweeps
MINOR_STEP_SIZE	106	2	uint16	Minor step bias voltage size in sweeps
MINOR_STEP_NO	108	2	uint16	Number of minor steps in sweeps

¹ During the same power cycle as the probe cassette release these fields will indicate the release times [/5 ms].

Approximate every other seconds the software read voltages and currents measured by the Hercules internal ADC. The values are stored in the memory, (as board.measured).

At the same time the status of the antennas are detected. The DEPLOY_1_TIME and DEPLOY_2_TIME are 1 if the antennas on boom cassette 1 and 2 are released, and 0 if they are stowed.

The housekeeping data are then read manually with the *0x40 - Get Housekeeping Information* command, or read automatically in science mode. No measurements are done when the housekeeping data are read.

BIAS1_VOLTAGE, BIAS2_VOLTAGE, BIAS3_VOLTAGE, BIAS4_VOLTAGE The 16 bit values (b) from the housekeeping are converted to probe bias values in volt (*bias_volt*) with the formulae:

$$bias_volt = b/1000 - 10$$

The TEMP1_THERMISTOR and TEMP2_THERMISTOR are the measured thermistor values in millivolt. This function returns the BRIK-II mNLP thermistor temperature in °C as function of the measured thermistor voltage in millivolt. Note that the readings are not calibrated, and with a low resolution of about 1°C, and should only be used as an indication of actual temperatures.

```
#define THERMISTOR_SERIES_RESISTANCE 10E3
#define THERMISTOR_MEASURING_VOLTAGE 3.3
#define K_ZERO 273.15
#define SHC_A 8.781626E-04
#define SHC_B 2.531972E-04
#define SHC_C 1.840754E-07
```

```
double Millivolt2Temperature(double millivolts_measured)
```

```

{
    double temperature; // temperature measured
    double current; // current through thermistor
    double resistance; // resistance of thermistor
    double rr; // log resistance of thermistor

    current = ((THERMISTOR_MEASURING_VOLTAGE - (millivolts_measured / 1000.0)) /
        THERMISTOR_SERIES_RESISTANCE);

    resistance = ((millivolts_measured / 1000.0) / current);

    rr = log(resistance);
    temperature = (SHC_A + (SHC_B * rr) + (SHC_C * rr * rr * rr));
    temperature = (1.0 / temperature);
    temperature = (temperature - K_ZERO);
    return(temperature);
}

```

The instrument has to be in science mode for the ± 12 V and probe bias measurements to be correct.

bit	BOARD_STATUS	Explanation
b0 (LSB)	LTC2344 CH1 status	status byte of LTC2344 ADC CH1 not equal 0x07
b1	LTC2344 CH2 status	status byte of LTC2344 ADC CH2 not equal 0x0F
b2	LTC2344 CH3 status	status byte of LTC2344 ADC CH3 not equal 0x17
b3	LTC2344 CH4 status	status byte of LTC2344 ADC CH4 not equal 0x1F
b4	SWEEP_CH1	probe bias sweep on ch1
b5	SWEEP_CH2	probe bias sweep on ch2
b6	RELEASE_KEY_ARM	The 'Start Boom Deployment' with KEY equal ARM was detected by the software.
b7	RELEASE_KEY_RELEASE	The 'Start Boom Deployment' with KEY equal RELEASE was detected by the software.
b8	RELEASE_PREV_ARM	The previous command was KEY equal ARM.
b9	RELEASE_TIME_OK	The RELEASE was sent after ARM within the time limit.
b10	RELEASE_2B	The boom cassette 2B is released.
b11	RELEASE_2A	The boom cassette 2A is released.
b12	RELEASE_1B	The boom cassette 1B is released.
b13	RELEASE_1A	The boom cassette 1A is released.
b14	ALREADY_RELEASED_2	The Boom Cassette 2 was already released.
b15	ALREADY_RELEASED_1	The Boom Cassette 1 was already released
b16	VERIFY_P1V2_VOLT	checking if the 1.2V has correct voltage
b17	VERIFY_P12V_VOLT	checking if the 12V has correct voltage

b18	VERIFY_N12V_VOLT	checking if the -12V has correct voltage
b19	VERIFY_BIAS_CH1	checking if the bias ch1 has correct voltage
b20	VERIFY_BIAS_CH2	checking if the bias ch2 has correct voltage
b21	VERIFY_BIAS_CH3	checking if the bias ch3 has correct voltage
b22	VERIFY_BIAS_CH4	checking if the bias ch4 has correct voltage
b23	VERIFY_P3V3_CURRENT	checking if the 3.3V has the correct current
b24	VERIFY_EGUN_P5V_CURRENT	checking if the Egun 5V has correct current
b25	VERIFY_P5V_CURRENT	checking if the 5V has correct current
b26	VERIFY_P1V2_CURRENT	checking if the 1.2V has the correct current
b27	SWEEP_CH3	probe bias sweep on ch3
b28	SWEEP_CH4	probe bias sweep on ch4
b29	PPS_PULSE_ARRIVED	a PPS pulse has been detected by the software
b30	BOARD_ID_1	value of Board ID jumper 1 (hetPort1[20])
b31(MSB)	I2C_ADDR_1	value of I2C address jumper 1 (canbus2 Rx)

The default Board ID is 1. When the BOARD_ID_1 (jumper) is 1, the application software set the Board ID to 2 .

The default I2C slave address is 0x49. When the I2C_ADDR_1 (jumper) is 1, the application software set the I2 slave address to 0x36.

On the Bartolomeo mNLP BUS3 boards the BOARD_ID_1 (jumper) is 0 and I2C_ADDR_1 (jumper) is 0. (BUS3: Board ID is 1 and I2C slave address is 0x49).

On the Bartolomeo mNLP BUS4 boards the BOARD_ID_1 (jumper) is 1 and I2C_ADDR_1 (jumper) is 1. (BUS4: Board ID is 2 and I2C slave address is 0x36).

Usage Example:

Transfer the following bytes to the system.

```
{<STID> <IVID> <BID> 0x40}
```

2.9 0x62 (0x63) – Get Configuration Parameters

Write length: 8 bytes.

Read length: 9 bytes + PARSZ parameter bytes.

Action: Allows reading configuration parameters.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
PARIDX	4	2	uint16	parameter start index
PARSZ	6	2	uint16	parameter byte length to return

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.
PARIDX	5	2	uint16	parameter start index
PARSZ	7	2	uint16	parameter byte length to return
BIAS1_VOLTAGE	9	2	uint16	Bias voltage probe 1
BIAS2_VOLTAGE	11	2	uint16	Bias voltage probe 2
BIAS3_VOLTAGE	13	2	uint16	Bias voltage probe 3
BIAS4_VOLTAGE	15	2	uint16	Bias voltage probe 4
SAMPLING_RATE	17	1	uint8	AD7768 sampling rate (lookup table index)
GAIN	18	1	uint8	Gain in pre-amplifiers [1,10,20]
PACKET_CONTENT	19	1	uint8	Content of Science data packet
NE_MAX	20	1	uint8	Maximum electron density
SAMPLING_FREQ_LTC2344	21	2	uint16	LTC2344 sampling frequency in fixed bias mode [Hz]

HK_RATE	23	2	uint16	Science packet to Housekeeping packet rate
EGUN_FIL1_ON	25	1	uint8	Electron emitter (EGUN) filament 1 on/off (1/0)
EGUN_FIL2_ON	26	1	uint8	Electron emitter filament 2 on/off (1/0)
EGUN_FIL1_V	27	2	uint16	Electron emitter filament 1 voltage [mV]
EGUN_FIL2_V	29	2	uint16	Electron emitter filament 2 voltage [mV]
EGUN_FIL_VMAX	31	2	uint16	Electron emitter filament voltage maximum [mV]
EGUN_ANODE_V	33	1	uint8	Electron emitter acceleration potential, lookup table (see below)
EGUN_AUTO	34	1	uint8	Not used
FILAMENT_ENABLE_CODE	35	2	uint16	Electron emitter (EGUN) filament enable code
EGUN_GRID_V	37	2	uint16	Electron emitter grid voltage, lookup table (see below)
BAD_CHANNEL_IND	39	1	uint8	Dismiss data from channels
BOOM_RELEASE_COUNT	40	1	uint8	Maximum boom release count. The release pulse is maximum BOOM_RELEASE_COUNT * 5 [ms]
OVERRIDE_P2V5	41	1	uint8	Override p2V5 reference calibration in science mode.
RS422_BAUD_RATE	42	1	uint8	RS422 baud rate
PROBE_WASH	43	2	uint16	Probe wash setup
CAL_CH_G1	45	16	uint16	Channel calibration data with gain set to 1
CAL_CH_G10	61	16	uint16	Channel calibration data with gain set to 10
CAL_CH_G20	77	16	uint16	Channel calibration data with gain set to 20
CAL_CH_BIAS	93	16	uint16	Channel bias calibration data

The PARIDX, parameter start index, is 0 on Offset 9, BIAS1_VOLTAGE .
The PARIDX is then the "Offset [Bytes]" - 9.

The PARSZ is the length of the parameters in bytes.

In order to get only BIAS1_VOLTAGE the PARIDX is 0 and PARSZ is 2.

In order to get all parameters the PARIDX is 0 and PARSZ is 102.

Usage Example:

Transfer the following bytes to the system.

```
{<STID> <IVID> <BID> 0x62 0x00 0x00 0x00 0x08}
```

The system returns a 17 byte response containing a 8 byte block starting from parameter index 0, (BIAS1_VOLTAGE).

2.10 0x64 (0x65) – Set Configuration Parameters

Write length: 8 bytes + PARSZ parameter bytes.

Read length: 5 bytes.

Action: Allows writing configuration parameters.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
PARIDX	4	2	uint16	parameter start index
PARSZ	6	2	uint16	parameter byte length
BIAS1_VOLTAGE	8	2	uint16	Bias voltage probe 1
BIAS2_VOLTAGE	10	2	uint16	Bias voltage probe 2
BIAS3_VOLTAGE	12	2	uint16	Bias voltage probe 3
BIAS4_VOLTAGE	14	2	uint16	Bias voltage probe 4
SAMPLING_RATE	16	1	uint8	AD7768 sampling rate (lookup table index)
GAIN	17	1	uint8	Gain in pre-amplifiers (default 1 , 10 or 20)
PACKET_CONTENT	18	1	uint8	Content of Science data packet
NE_MAX	19	1	uint8	Maximum electron density
SAMPLING_FREQ_LTC2344	20	2	uint16	LTC2344 ADC sampling frequency in fixed bias mode [Hz]
HK_RATE	22	2	uint16	Science packet to Housekeeping packet rate
EGUN_FIL1_ON	24	1	uint8	Electron emitter filament 1 on/off (1/0)
EGUN_FIL2_ON	25	1	uint8	Electron emitter filament 2 on/off (1/0)

EGUN_FIL1_V	26	2	uint16	Electron emitter filament 1 voltage [mV]
EGUN_FIL2_V	28	2	uint16	Electron emitter filament 2 voltage [mV]
EGUN_FIL_VMAX	30	2	uint16	Electron emitter filament voltage maximum [mV]
EGUN_ANODE_V	32	1	uint8	Electron emitter acceleration potential, lookup table (see below)
EGUN_AUTO	33	1	uint8	Not used
FILAMENT_ENABLE_CODE	34	2	uint16	EGUN Filament enable code
EGUN_GRID_V	36	2	uint16	Electron emitter grid voltage, lookup table (see below)
BAD_CHANNEL_IND	38	1	uint8	Dismiss data from channels
BOOM_RELEASE_COUNT	39	1	uint8	Maximum boom release count. The release pulse is maximum BOOM_RELEASE_COUNT * 5 [ms] long. Default is 0.
OVERRIDE_P2V5	40	1	uint8	Override p2V5 reference calibration in science mode.
RS422_BAUD_RATE	41	1	uint8	RS422 baud rate
PROBE_WASH	42	2	uint16	Probe wash setup

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

The PARIDX, parameter start index, is 0 on Offset 8, BIAS1_VOLTAGE .
The PARIDX is then the "Offset [Bytes]" - 8.

The PARSZ is the length of the parameters in bytes.

In order to send only BIAS1_VOLTAGE the PARIDX is 0 and PARSZ is 2.

In order to send all parameters the PARIDX is 0 and PARSZ is 36.

PROBE BIAS:

BIAS1_VOLTAGE, BIAS2_VOLTAGE, BIAS3_VOLTAGE, BIAS4_VOLTAGE

To set the bias value on a probe to bias_volt value in volt
the 16 bit value (*b*) sent to the instrument is:

$$b = (int)((bias_volt * 1000) + 10000);$$

AD7768 ADC:

The AD7768 SAMPLING_RATE are:

SAMPLING_RATE	Sampling rate [Hz]	f_internal	dec_rate
51 (0x33)	9765.625	2.5 MHz	256
52 (0x34)	4882.8125	2.5 MHz	512
53 (0x35)	2441.40625	2.5 MHz	1024
37 (0x25)	1220.703125	1.25 MHz	1024
4 (0x04)	610.3515625	312.5 kHz	512
5 (0x05)	305.17578125 (default)	312.5 kHz	1024
6 (0x06)	152.587890625	312.5 kHz	1024
1 (0x01)	4882.8125	312.5 kHz	64
2 (0x02)	2441.40625	312.5 kHz	128

The SAMPLING_RATE is the parameter byte sent to the instrument.

The Sampling rate [Hz] is the output data rate (ODR) from the instrument.

The f_internal is the intern sampling rate inside the AD7768 ADC.

The dec_rate is the number of data samples averaged inside the AD7768 ADC.

The 152 Hz is an average of two samples in the flight software (firmware) from the 305 Hz.

If the 9765.625 Hz is selected, the RS422_BAUD_RATE parameter byte also has to be set to 92 (921600 baud).

LTC2344 ADC:

The LTC2344 ADC must have been selected and running in science mode in order to set the LTC2344 sampling frequency. The default sample frequency is 1000 Hz. The minimum sample frequency is 154 Hz, and the maximum sample frequency is 3130 Hz.

EGUN:

The mNLP instrument has to be in science mode in order to set the electron emitter. To set the electron emitter the FILAMENT_ENABLE_CODE has to be 0xE11E. Only set the FILAMENT_ENABLE_CODE when the mNLP instrument is in vacuum.

The EGUN_FIL1_ON and EGUN_FIL2_ON are set to 1 to turn the emitters on. If the FILAMENT_ENABLE_CODE is not equal 0xE11E, only the emitter electronic board is switch on, and not the emitter filaments. The EGUN_FIL1_V and EGUN_FIL2_V are the millivolt applied to the emitter filaments. They are not recommended to be set larger than 1200 mV.

The values for the EGUN acceleration and grid voltages are (*on emitter board test A (not FM) and instrument BTL V4 BUS3 A*):

EGUN_ANODE_V	Acceleration voltage [V]	EGUN_ANODE_V	Acceleration voltage [V]
0 (default)	-61.53	50	-108.94
10	-71.17	60	-118.37
20	-80.58	70	-127.7
30	-90.26	80	-137.08
40	-99.48	90	-146.51

EGUN_ANODE_V = 0

EGUN_GRID_V	Grid voltage [V]	EGUN_GRID_V	Grid voltage [V]
0 (default)	-64.18	70	-75.85
10	-65.93	80	-77.55
20	-67.62	90	-79.25
30	-69.27	100	-81.03
40	-70.9	110	-82.92
50	-72.55	120	-84.87
60	-74.2	130	

Grid voltage [V] – Acceleration voltage [V] at EGUN_ANODE_V = 0

EGUN_GRID_V	Grid voltage [V]	EGUN_GRID_V	Grid voltage [V]
0 (default)	-2.65	70	-14.32
10	-4.4	80	-16.02
20	-6.09	90	-17.72
30	-7.74	100	-19.5
40	-9.37	110	-21.39
50	-11.02	120	-23.34
60	-12.67	130	

RS422 BAUD RATE:

The default baud rate is 921600 .

When RS422_BAUD_RATE is 92 (0x5C) it is set to 921600.

When RS422_BAUD_RATE is 46 (0x2E) it is set to 460800.

When RS422_BAUD_RATE is 23 (0x17) it is set to 230400.

PROBE WASH:

When the instrument is in idle mode and the PROBE_WASH is set to 0xB97C, the probe wash routine is run. Then a voltage of 94.57 VDC (*on Brik-II ScinMon M7*) is applied on the probes. This voltage is on until a new PROBE_WASH with a value different from 0xB97C arrives.

Usage Example:

Transfer the following bytes to the system.

```
{<STID> <IVID> <BID> 0x64 0x00 0x00 0x00 0x08 0x27 0x10 0x27 0x10 0x27
0x10 0x27 0x10}
```

The system updates 8 parameter bytes to [0x27 0x10 0x27 0x10 0x27 0x10 0x27 0x10] starting from parameter index 0, (BIAS1_VOLTAGE). The probe bias values are set to 0 volt.

2.11 0x66 (0x67) – Set PPS Timestamp

Write length: 9 bytes.

Read length: 5 bytes.

Action: Send time when PPS pulse was transmitted.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
PPS_TIMESTAMP	4	4	uint32	Timestamp (Unix/Linux epoch)
APPLICABILITY	8	1	uint8	Timestamp valid: On message arrival 0x00

				With next PPS pulse 0xFF
--	--	--	--	--------------------------

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

The UPTIME is the time in milliseconds since mNLP transitioned from Startup to Idle state. When a PPS pulse arrives the UPTIME is stored in the UPTIME_PPS_TIMESTAMP variable.

The PPS pulse can be a 3.3V signal, which is set to 3.3 V for 0.5 milliseconds.

The UiO GSE sets the GPIO_B_00 line to 0.0 V when the GSE starts. Then it sets the GPIO_B_00 line to 0.0 V when the GSE programs stops. A 20 kΩ resistor is connected between the GPIO_B_00 output and the mNLP PPS input.

When the APPLICABILITY is 0x01 the PPS pulse was sent before the *Set PPS Timestamp* command. The UPTIME_PPS_TIMESTAMP variable is set when the PPS pulse is detected.

When the APPLICABILITY is 0xFF the PPS pulse is sent after the *Set PPS Timestamp* command. The UPTIME_PPS_TIMESTAMP variable is set when the PPS pulse is detected.

When the APPLICABILITY is 0x00 no PPS pulse is available. The UPTIME_PPS_TIMESTAMP variable is set when the *Set PPS Timestamp* command is received and executed in the mNLP software.

When the PPS pulse is transmitted and the *Set PPS Timestamp* command is sent, the time of the housekeeping packet is $PPS_TIMESTAMP + UPTIME - UPTIME_PPS_TIMESTAMP + 'offset'$.

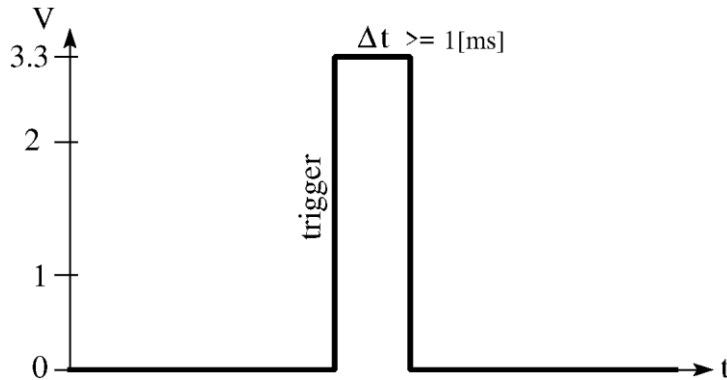


Figure 2.3 This figure shows the PPS pulse.

The PPS signal line is at 0 [V]. When the PPS signal appears the PPS interrupt function trigger at the positive edge, and the mNLP time is set

2.12 0xF0 (0xF1) – Request Debug Message

Write length: 4 bytes

Read length: 5 + 64 bytes.

Action: Poll for debug messages

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.
DEBUG_MESSAGE	5	64	uint8	Debug message terminated by '\0'

When DEBUG flag in application is set, it is possible to poll debug messages from the application via I2C. The application will only hold one debug message at the time.

If the reply contains a new unread debug message. The STAT byte for this message will also have the DEBUG_MESSAGE_INDICATION_BIT (0x40) set.

The *Request Debug Message* command request for one (the last) debug message.

2.13 0xF6 (0xF7) – Request Calibration Update

Write length: 132 bytes

Read length: 5 bytes.

Action: Allows writing configuration parameters and calibration data to NVM.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
KEY	4	3	uint24	0xBEEF33 Must be correct for command to be accepted
BIAS1_VOLTAGE	7	2	uint16	Bias voltage probe 1
BIAS2_VOLTAGE	9	2	uint16	Bias voltage probe 2
BIAS3_VOLTAGE	11	2	uint16	Bias voltage probe 3
BIAS4_VOLTAGE	13	2	uint16	Bias voltage probe 4
SAMPLING_RATE	15	1	uint8	Sampling rate (lookup table index)
GAIN	16	1	uint8	Gain in pre-amplifiers
PACKET_CONTENT	17	1	uint8	Content of Science data packet
NE_MAX	18	1	uint8	Maximum value for Ne calculation
RAD_RATE	19	2	uint16	Science packet to Radiation packet rate
HK_RATE	21	2	uint16	Science packet to Housekeeping packet rate
EGUN_FIL1_ON	23	1	uint8	Electron emitter filament 1 on/off
EGUN_FIL2_ON	24	1	uint8	Electron emitter filament 2 on/off
EGUN_FIL1_V	25	2	uint16	Electron emitter filament 1 voltage
EGUN_FIL2_V	27	2	uint16	Electron emitter filament 2 voltage
EGUN_FIL_VMAX	29	2	uint16	Electron emitter filament voltage maximum
EGUN_ANODE_V	31	1	uint8	Electron emitter acceleration potential
EGUN_AUTO	32	1	uint8	Electron emitter autonomous mode 0:off, 1: based on raw data or

				2: based on calculated potential
EGUN_VPOR_U	33	2	uint16	EGUN upper limit input for autonomous mode
EGUN_VPOR_L	35	2	uint16	EGUN lower limit input for autonomous mode
BAD_CHANNEL_INDICATION	37	1	uint8	Program exception bits
RADMON_CORE_MINIMUM	38	1	uint8	Minimum voltage on RADMON RAM
RADMON_MODE	39	2	uint16	Radiation monitoring setup
PROBE_WASH	41	2	uint16	Probe wash setup
CAL_CH_G1	43	16	uint8	Channel calibration data with gain set to 1
CAL_CH_G10	59	16	uint8	Channel calibration data with gain set to 10
CAL_CH_G20	75	16	uint8	Channel calibration data with gain set to 20
CAL_CH_BIAS	91	16	uint8	Channel bias calibration data

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

This command is only accepted by the bootloader, and is used to write calibration data and default configuration to FLASH memory. The data is loaded to RAM by the application every time the application is started.

(The RAD_* and RADMON_* parameters are not used by the Bartolomeo mNLP software.)

2.14 0x57 (0x58) – Run Bias Sweep

Write length: 28 bytes

Read length: 5 bytes.

Action: Sweep the probe bias.

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header
SWEEP_MODE	4	2	uint16	Sweep mode
SAMPLE_FREQ	6	2	uint16	Sample frequency in Hz
STEP_LENGTH_TIME	8	2	uint16	Step length in unit of 0.1 us (fast sweep) or 0.01792 ms (slow sweep)
SAMPLE_HOLD_TIME	10	2	uint16	Sample hold time in unit of 0.1 us (fast sweep) or 0.01792 ms (slow sweep)
SWEEP_NO	12	4	uint32	Number of sweeps
BIAS_START_VOLTAGE	16	2	uint16	Bias probe voltage start
MAJOR_STEP_SIZE	18	2	uint16	Major step bias voltage size
MAJOR_STEP_NO	20	2	uint16	Number of major bias steps
MINOR_SWEEP_START	22	2	uint16	The minor sweep starts after # major steps
MINOR_STEP_SIZE	24	2	uint16	Minor step bias voltage size
MINOR_STEP_NO	26	2	uint16	Number of minor steps

Before running the bias sweeps the science mode must be started with the LTC2344 ADC.

The probe bias sweep steps the bias values upwards starting at BIAS_START_VOLTAGE [V] in MAJOR_STEP_NO number of upward steps. Each step is MAJOR_STEP_SIZE [V] increase in bias value. The step length time in seconds is STEP_LENGTH_TIME * [0.1 μ s] in fast mode, or STEP_LENGTH_TIME * [0.01792 ms] in slow mode. The STEP_LENGTH_TIME maximum number is 65535.

The sweep can also contain a minor sweep with smaller step size in the most important part of the sweep. This minor sweep starts at the MINOR_SWEEP_START major bias step. When MINOR_STEP_NO is 0 we only run the MAJOR sweep.

The SWEEP_NO is the number of sweeps run before the sweep finish, and the instrument returns to fixed bias science mode. The last sweep run only the first steps where the bias values are less than 0 volt. Approximate one second after the sweep finish the parameters sent by the last *Set Configuration Parameters* command are executed once more. The probe bias and the other settings will then be the values before the sweep was started.

The SAMPLE_FREQ is the LTC2344 ADC sample frequency in Hz. The minimum sample frequency is 154 Hz, and the maximum sample frequency is 3130 Hz.

The bias change create some noise in the data from the probes. The length of this disturbance is dependent on the step size in volt, MAJOR_STEP_SIZE [V]. When entering a new bias step, the ADC stop sampling in the interval SAMPLE_HOLD_TIME, (in units of 0.1 μ s or 0.01792 ms).

The minimum step length time is (usually) 1.0 ms, STEP_LENGTH_TIME equal 10000 in fast mode. The minimum sample hold time can be seen in the data or measured on an oscilloscope.

The maximum total number of steps up and down is 401.

When a sweep command with MAJOR_STEP_NO equal 0, the current sweep stops.

The bias values are encoded in the same manner as in the *Set Configuration Parameters* command. To set the bias value on a probe to bias_volt value in volt, the 16 bit value (b) sent to the instrument is: $b = (int)((bias_volt * 1000) + 10000);$

SWEEP_MODE , (default is 270 , 0x00010E , binary 0000 0001 0000 1110)

Bit	Name	Hex	Value 0 (off)	Value 1 (on)
0	sweep ch1	0001	do not sweep on probe1	bias sweep on probe1
1	sweep ch2	0002	do not sweep on probe2	bias sweep on probe2
2	sweep ch3	0004	do not sweep on probe3	bias sweep on probe3
3	sweep ch4	0008	do not sweep on probe4	bias sweep on probe4
4	up down	0010	upsweep and down-sweep	only upsweep
5	fast or slow	0020	step size resolution 0.1 μ s	step size resolution 0.01792 ms
6	x10	0040	no extra clock division	step size resolution * = 10 This option is not implemented.
7	bias packet	0080	no bias sweep packets	send bias sweep packets
8	replace ch1	0100	probe1 data in ch1	encoded bias values in ch1
9	replace ch2	0200	probe2 data in ch2	encoded bias values in ch2
10	replace ch3	0400	probe3 data in ch3	encoded bias values in ch3 The ch3 is the LTC2344 CH1. The firmware does not work when replacing ch3 data with the bias values.
11	replace ch4	0800	probe4 data in ch4	encoded bias values in ch4
12	invert ch1	1000	do not invert ch1	multiply bias with -1.0 on ch1
13	invert ch2	2000	do not invert ch2	multiply bias with -1.0 on ch2
14	invert ch3	4000	do not invert ch3	multiply bias with -1.0 on ch3
15	invert ch4	8000	do not invert ch4	multiply bias with -1.0 on ch4

when bit 0, 1, 2 and 3 are 0 , sweep on all channels

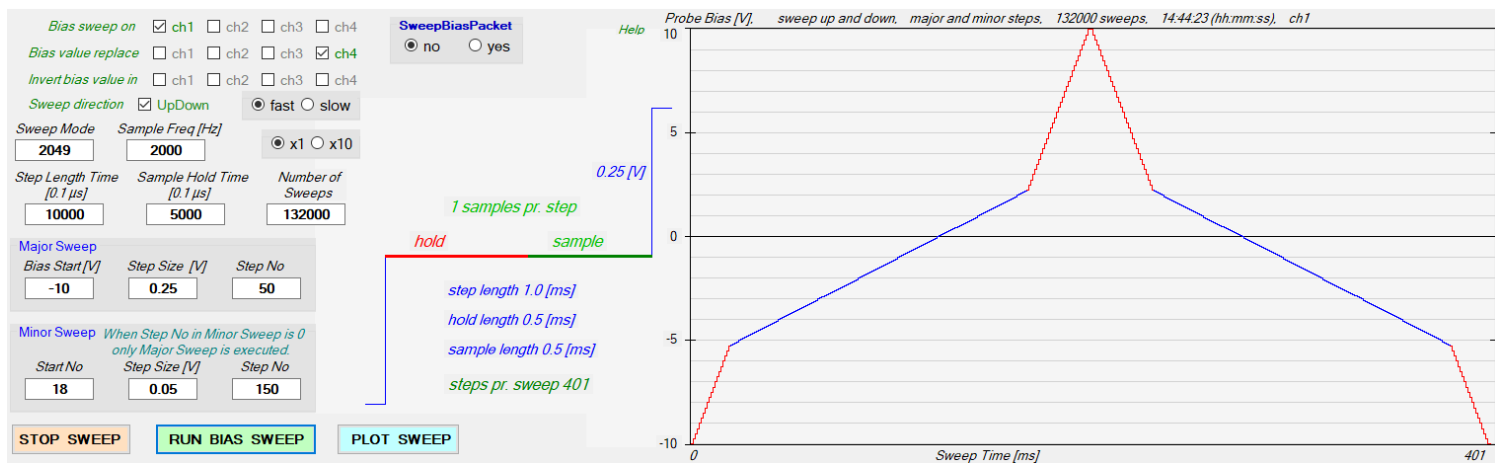


Figure 2.4 This figure shows the UiO GSE sweep interface.

To stop the bias sweeping a Run Bias Sweep with all parameters equal 0 is sent to the mNLP instrument.

Type	Step Length Time	Bias Step Length
fast	10000	1 ms
fast	50000 (default)	5 ms
fast	65535 (maximum)	6.5535 ms
slow	558	10 ms
slow	5580	100 ms
slow	55804	1 s
slow	65535 (maximum)	1.174 s

Reply:

Name	Offset [Bytes]	Size [Bytes]	Type	Description
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.

2.15 0x45 (0x46) – Get Temperature

Write length: 4 bytes

Read length: 9 bytes.

Action: Poll for debug messages

Command:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
CC	3	1	uint8	See code in header

Reply:

<i>Name</i>	<i>Offset [Bytes]</i>	<i>Size [Bytes]</i>	<i>Type</i>	<i>Description</i>
STID	0	1	uint8	System type ID.
IVID	1	1	uint8	Interface version ID.
BID	2	1	uint8	Board ID.
RC	3	1	uint8	Response code as stated in the header (inside parenthesis).
STAT	4	1	uint8	Command acceptance status and new indicator.
TEMP1_TERMISTOR	5	2	uint16	Thermistor 1 temperature [°C] (encoded)
TEMP2_TERMISTOR	7	2	uint16	Thermistor 2 temperature [°C]

This command is equal to the *'0x40 (0x41) – Get Housekeeping Information'* command. However, only the thermistor temperatures are returned. See Sect. 2.8 for how to calculate the temperature in celcius [°C] from the 16-bit values.

3 Science Data Interface

The data from the mNLP instrument on the serial line is packet based. The default baud rate is 921600.

3.1 Science packet (1024 bytes)

0xa1	1byte PACKET_CONTENT	3 bytes uptime	1016 bytes science data ch1, ch2, ch3, ch4, ch1, ...	2 byte CRC	0x1a
------	-------------------------	-------------------	--	---------------	------

The 1016 byte science data contains 127, 16-bit signed raw data samples from ch1, ch2, ch3 and ch4. When running bias sweep one of the channels may contain the encoded probe bias.

The 3 bytes uptime in the science packet is the time in milliseconds since mNLP transitioned from Startup to Idle.

The PACKET_CONTENT byte:

Bit	Hex	Value
0 (LSB)	0x01	ch1 raw data
1	0x02	ch2 raw data
2	0x04	ch3 raw data
3	0x08	ch4 raw data
4	0x10	0: fixed bias mode , 1: sweep is running
5	0x20	0: AD7768 ADC, 1: LTC2344 ADC
6	0x40	Board ID (LSB)
7 (MSB)	0x80	Board ID (MSB)

3.2 Housekeeping packet (109 bytes)

0xb2	105 bytes housekeeping data, (110 - 5(STID, IVID, BID, RC, STAT))	2 byte CRC	0x2b
------	--	------------	------

The 81 bytes housekeeping data are explained in Sect. 2.8. The first byte is the MODE byte.

The MODE byte (in Housekeeping response and Housekeeping packet):

Bit	Hex	Value
0 (LSB)	0x01	mode (LSB)
1	0x02	mode , mode = { 0: <i>bootloader</i> , 1: <i>startup</i> , 2: <i>idle</i> ,
2	0x04	mode 3: <i>deploy</i> , 4: <i>science</i> }
3	0x08	mode (MSB)
4	0x10	when science { 0: fixed bias mode, 1: sweep is running }
5	0x20	when science { 0: AD7768 ADC, 1: LTC2344 ADC }
6	0x40	board ID (LSB)
7 (MSB)	0x80	board ID (MSB)

3.3 Clock packet (12 bytes)

0xd4	4 bytes seconds	4 bytes ns	2 byte CRC	0x4d
------	-----------------	------------	------------	------

The 4 bytes seconds are the number of seconds since 1970-01-01 00:00:00, (UNIX time). This packet is **not** coming from the mNLP instrument, but is created by the PC GSE. The time is the PC time. The use of this packet is to calibrate the time calculated by using the UPTIME, and PPS information in the housekeeping packet.

3.4 Bias sweep packets (1024 bytes)

0xe5	3 bytes uptime	2 bytes bias	3 bytes uptime	2 bytes bias		2 byte CRC	0x5e
------	----------------	--------------	----------------	--------------	-------	------------	------

0xe6	3 bytes uptime	2 bytes bias	3 bytes uptime	2 bytes bias		2 byte CRC	0x6e
------	----------------	--------------	----------------	--------------	-------	------------	------

When bit 7 in SWEEP_MODE is 1 and the mNLP instrument is sweeping, the optional bias sweep packet is sent.

It contains the 24-bit uptime and the 16-bit encoded bias value for each bias step. The bias sweep packet contains 204 encoded bias values.

The 0xe5, 0x5e is used for Board ID 0x01, see Sect 1.4.8.
The 0xe6, 0x6e is used for Board ID 0x02.

3.5 Science data volume

Each data sample is composed of 8 bytes when all probes are active. Using science data packets of 1024 bytes, with 8 bytes header/CRC, we have a overhead of 0,8% (excluding uart start and stop bits). See table below for typical expected data rates at selected sampling rates.

Sample rate	Data rate	Packets/s
1000	8kB/s	7.87
9765.625	78.5kB/s	76.89

Housekeeping packets are updated at <1Hz, and are not included in the data rate.

9765.625 Hz is maximum AD7768 sampling rate. 3130 Hz is maximum LTC2344 sampling rate.

