



Technical Note

Comparison of instrument commanding
on Herschel
Ken King & Sunil Sidher

Ref: SPIRE-RAL-NOT-001618
Issue: 1.1
Date: 6th June 2003
Page: 1 of 3

This note is issued indirectly in response to an action from the HGSSE to investigate the methods used by the three Herschel instruments to command their instrument and the impacts these may have on the HGS system design.

Below we report on our understanding of the operation of each of the instruments. For clarity, we use the following terms:

- **Telecommand:** the command sent to the instrument in the form of a TC packet.
- **Mnemonic:** the name of a command used in the CUS. The mnemonic + arguments define the telecommand sent to the instrument
- **Command:** an instruction sent by the DPU/ICU to instrument internal subunits or subsystems

1. SPIRE

SPIRE executes telecommands in two ways:

Functions are telecommands, which are executed directly by code in the OBS. These are used to control OBS functions (manage on-board tables, set OBSID, BBID etc). They may take arguments, but the only way to change the functionality is to upload new OBS code.

Command Lists: the majority of the SPIRE operations are defined in this way. These are lists of instructions for an internal interpreter, which allows control of the commands sent to the instrument subsystems. Command Lists are executed using a TC(8,4) telecommand and may take input arguments. They may be compiled into the OBS and copied into memory at the startup of the OBS or loaded and updated using TC(8,4) telecommands.

2. HIFI

The HIFI OBS provides the same functionality as that of SPIRE, with the possibility to execute commands lists stored on board.

There are two types of telecommand:

Functions:

These telecommands are defined in the OBS code. They may take arguments passed in the TC packets that determine the actual commands sent to the HIFI subsystems. Some of these telecommands operate using feedback from the instrument to control flow of execution through the code. The code includes mechanisms to limit this (e.g. limiting the number of iterations that may be performed or waiting to ensure that an action has taken place) so that the maximum time of execution of a telecommand can be calculated.

There are additional telecommands to generate arbitrary single commands to the subsystems. These are needed during ground testing and will not be used after launch.

Configuration Commands:



Technical Note

Comparison of instrument commanding
on Herschel
Ken King & Sunil Sidher

Ref: SPIRE-RAL-NOT-001618
Issue: 1.1
Date: 6th June 2003
Page: 2 of 3

These telecommands redefine all the parameter values of a single subsystem through a set of arguments passed in the telecommand. The implementation of the configuration commands in the OBS requires some extra arguments in the telecommand (counters) to map the arguments in the telecommand packet onto the parameter list in the OBS.

Configuration Commands are implemented as command lists, compiled into the OBS code. The only way to change these is by patching with memory upload telecommands. As the list of subsystem parameters to set is of fixed length, the time to execute is fixed.

HIFI will only use the command lists (in the HIFI case referred to as 'virtual machine' programs) in cases where the timing requirements are such that they cannot be met in the OBSW.

3. PACS

There are four types of telecommands:

General PUS service types:

These are telecommands to implement the

- memory management (type 6)
- time management (type 9)
- TM packet transmission control (type 14)

Subsystem Commands:

These telecommands send a single command to a subsystem. They are sent as TM(8,4) telecommands with the Function_ID identifying the subunit addressed, the Activity_ID identifying the command to execute and the SID identifying the number of parameters to follow.

On Board Command Procedures (OBCP):

These are implemented as code (written in C). They usually contain no logic (although of course being written in C they can do anything), but allow some control of command timing (e.g. by providing steps to wait for a set period of time or wait for a flag to be set). They are usually, but not necessarily, used to execute one DEC sequence.

OBCP are controlled by using OBCP services (PUS type 18) but only a single OBCP can be loaded for testing purpose. All other OBCP are part of the DPU memory image and loaded using the memory management service (type 6). So during standard operations the behaviour of OBCP is defined by the loaded on-board software.

In order to establish the duration of OBCPs they will be coded as CUS scripts as well. Note however that CUS will generate the single telecommand "DPU_START_OBCP" to start the OBCP with the given parameters. Another useful outcome of this exercise is that the CUS user will benefit from a large visibility of the actions hidden under the OBCPs.

Note:

It is possible to execute an OBCP with a DEC sequence identifier passed as an argument, thus altering the function performed by the OBCP. This feature is not intended to be used during operations.

DEC Sequences:



Technical Note

Comparison of instrument commanding
on Herschel
Ken King & Sunil Sidher

Ref: SPIRE-RAL-NOT-001618
Issue: 1.1
Date: 6th June 2003
Page: 3 of 3

These are sets of commands executed in the DECMEC subsystem to be sent to further instrument subunits. They can contain some logic (e.g. loops).

DECMEC holds only a single DEC sequence. So every OBCP using a DEC sequence must load it before using it. For testing purpose standard function management telecommands (type 8,4) can be used to load and execute a DEC sequence.

Again standard DEC sequences are part of the onboard memory image controlled by the memory management telecommands.

4. CONCLUSIONS

The main concern of the Systems Engineering group is how to associate the function executed on board with the telecommand defined in the CUS and sent to the instrument.

For HIFI there is a one to one correspondence between telecommand sent to the instrument and the function executed (unless the OBS is changed).

For PACS and SPIRE this is not true as the function to be executed may be passed as an argument to the instrument telecommand. This makes the OBS more flexible for testing the operation of the instrument but could make it more difficult to associate the function and telecommand.

For SPIRE we intend to define specific mnemonics for each Command List in the HGS so that they will identify the function to be performed and as such will have a clearly defined set of arguments (and defined execution time). In this way the function of each command list will be fixed. However, we reserve the right, during testing, to have the ability to load a new command list and execute it, without having to create a new MIB and re-ingest it into the HCSS and SCOS. We can identify these command lists as 'Test Command Lists' and as such the function performed has to be determined from the content uploaded (they will never be compiled into the OBS). We will ensure that these will not be used in normal flight operations.