SpaceWire Protocol Analyzer on Space Cube®

Hiroki Hihara, **Masaharu Nomachi**
Shuichi Moriyama, Toru Tamura
Takayuki Tohma, Kenji Kitade
Tadayuki Takahashi, Takeshi Takashima
SpaceWire User's Group, Japan

Steve Parkes, and Stuart Mills
Univ. of Dundee/Star-DundeeLtd.

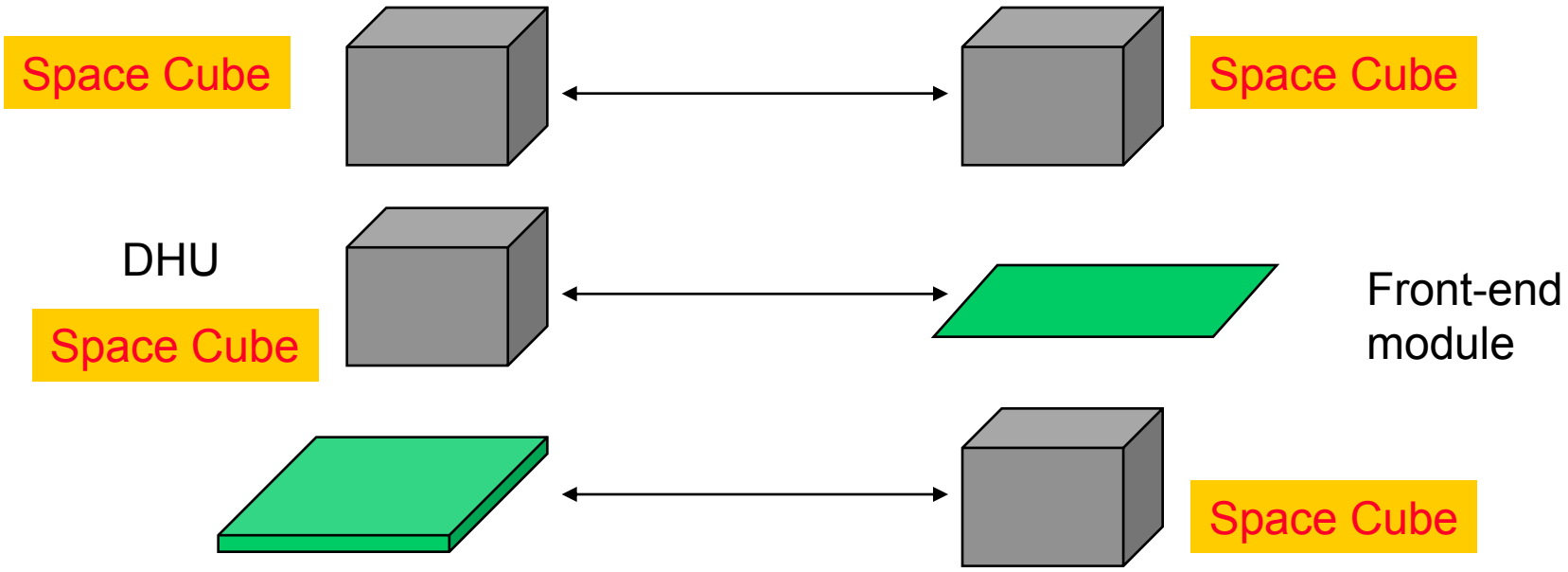
Introduction

- System development in large and geographically spread community
  - (ex) Joint project like Bepi-Colombo/MMO
  - **Modular system**
    - **share developments**
  - **A reference in the development process**

Simple and well defined interconnection
System emulation

❑ Ground test system

- DHU emulation
  - DHU software development, Front-end module test
- Front-end emulation
  - Test DHU, System integration test



Applications

- **Protocol tester**
  - USB brick (from Star Dundee)
  - Space Cube® (from SHIMAFUJI)
    - CUBA (Space Cube Analysis Software)
    - NTSpace (Japan) - University of Dundee (UK)

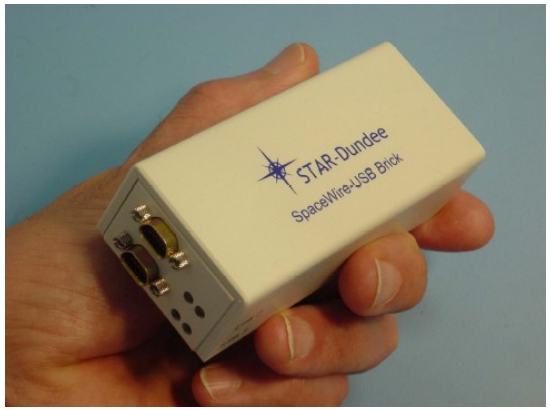
Space Cube is a stand-alone computer.

In case we need several **embedded** testers in a system, a stand alone one is preferable.

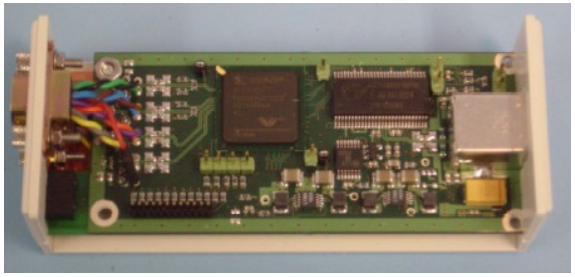
SpaceWire CUBA Software

- The same RMAP protocol analyzer software runs on both USB brick and Space Cube®

Common API is established

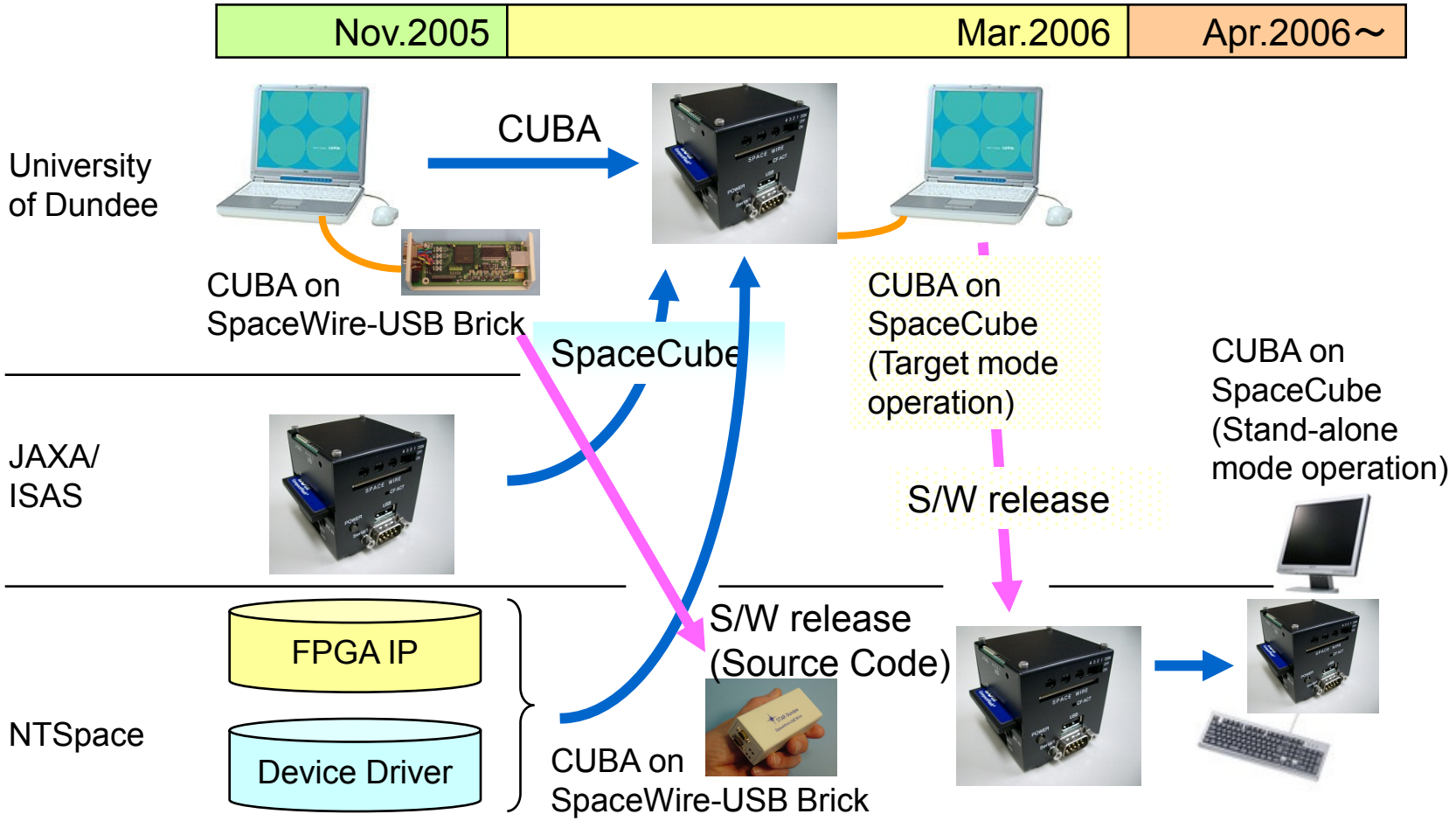


USB brick



SpaceCube®

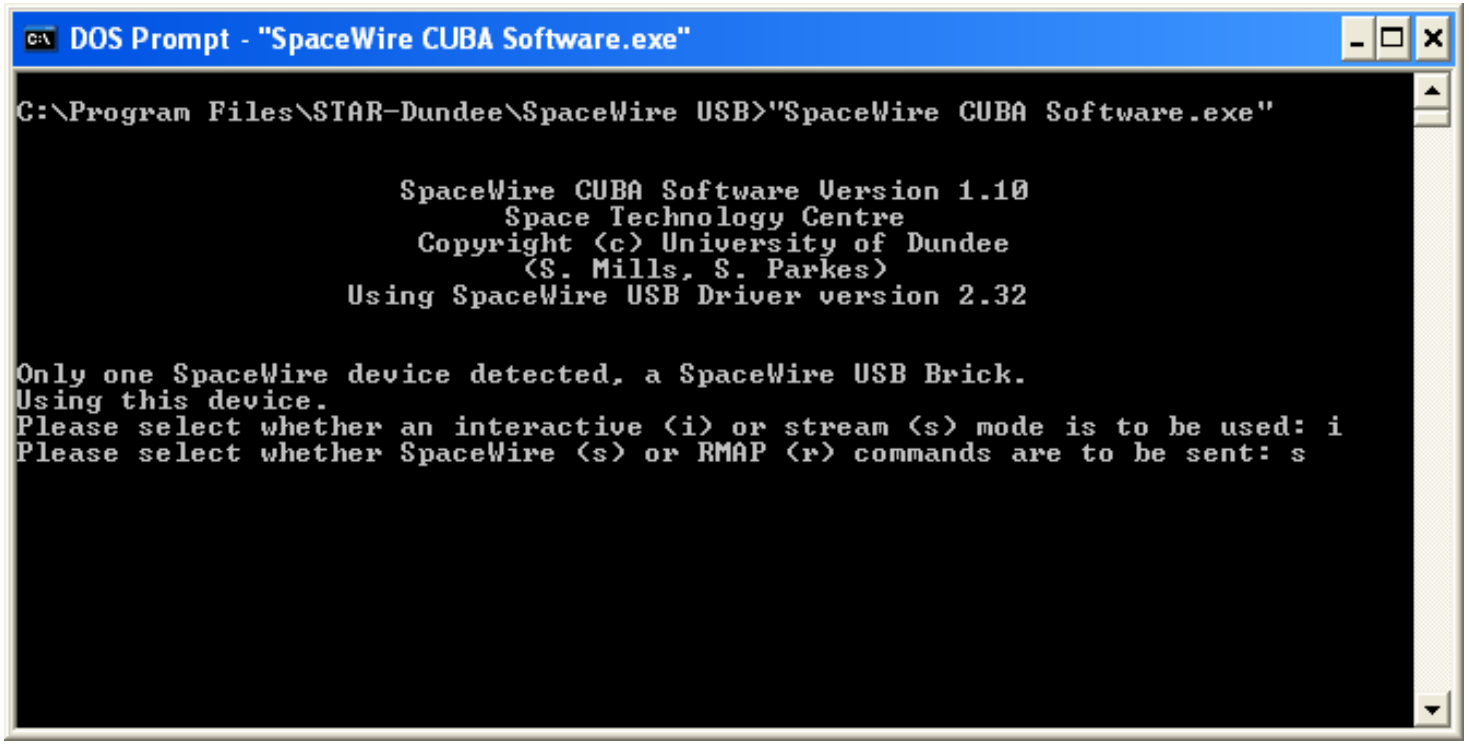
❑ Development History



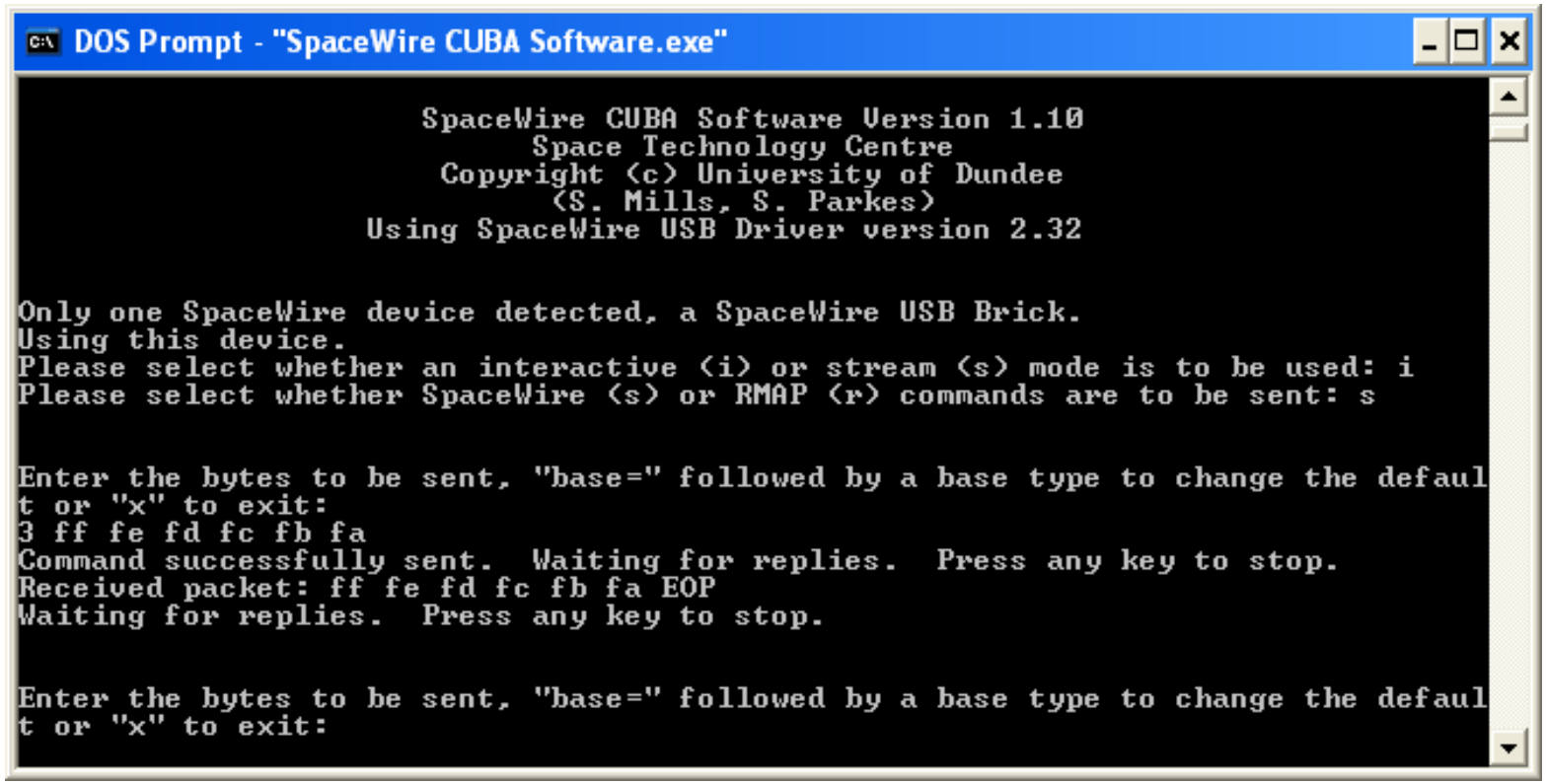
SpaceWire CUBA Software

- **Heritage of STAR-Dundee PETRI software**
  - Aimed at testing user equipment
  - Send commands, receive response
  - Commands defined in text file
  - When to receive defined in the text file
  - Run from command line
  - Responses put in text file
- **Four operation mode**
  - SpaceWire Interactive Mode
  - SpaceWire Command Stream Mode
  - RMAP Interactive Mode
  - RMAP Command Stream Mode

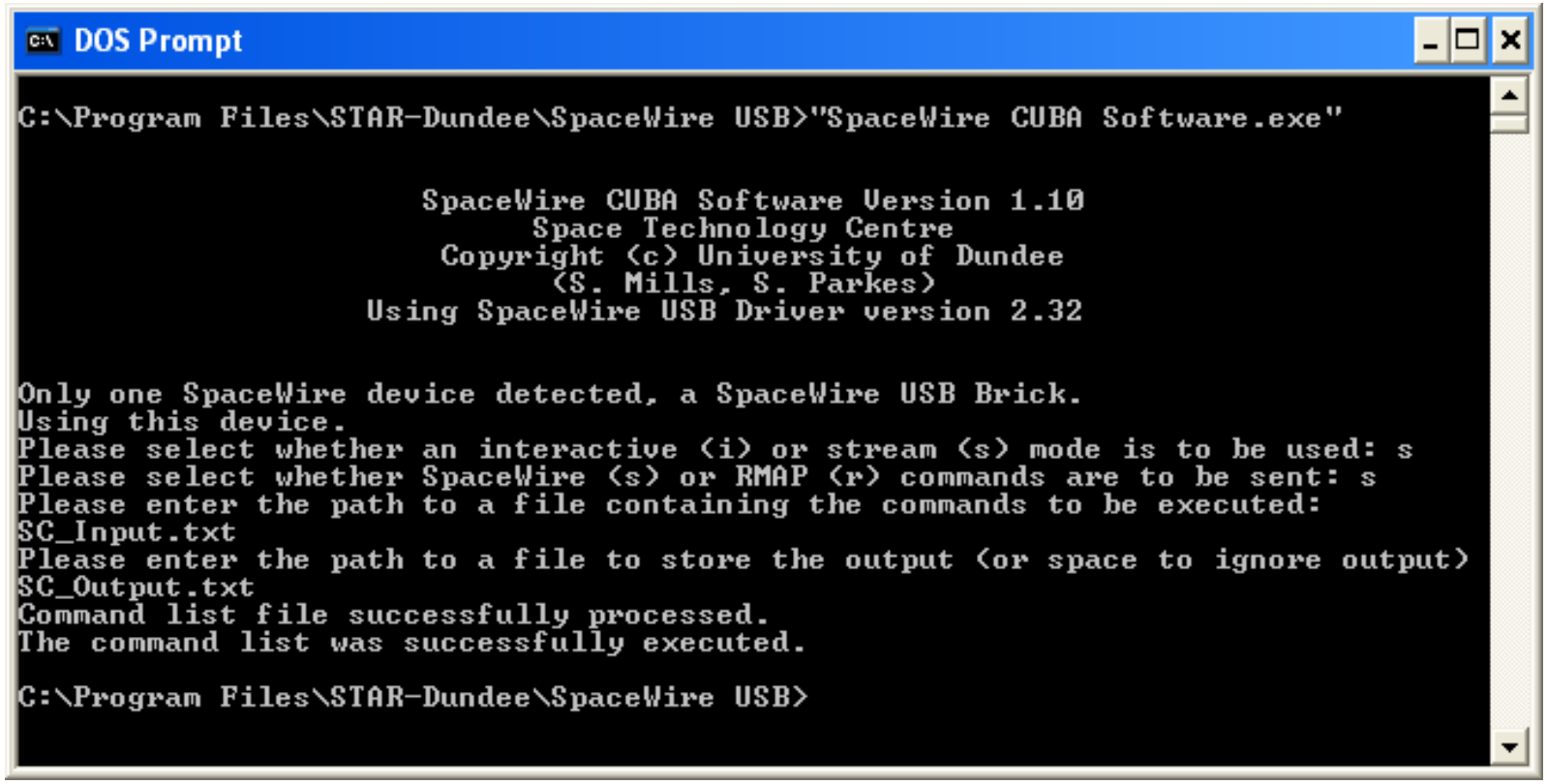
❑ Specify Mode



❑ SpaceWire Interactive Mode



❑ SpaceWire Command Stream Mode



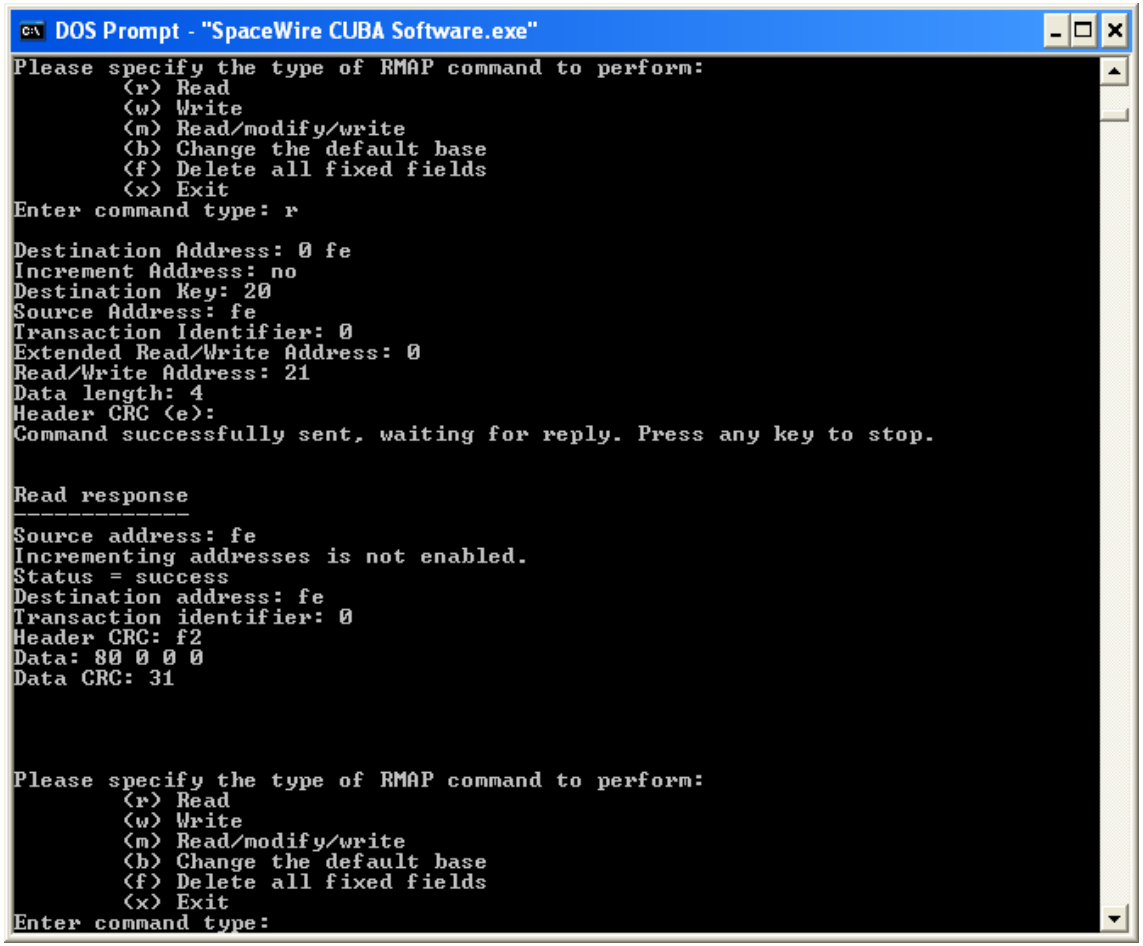
- **SpaceWire Command Stream Mode**
  - Example Input:

```
/* Send 2 packets */
S N:2 D:1 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff;

/* Change the input base to decimal */
B 10;

/* Receive 2 packets */
R N:2 L:16;
```

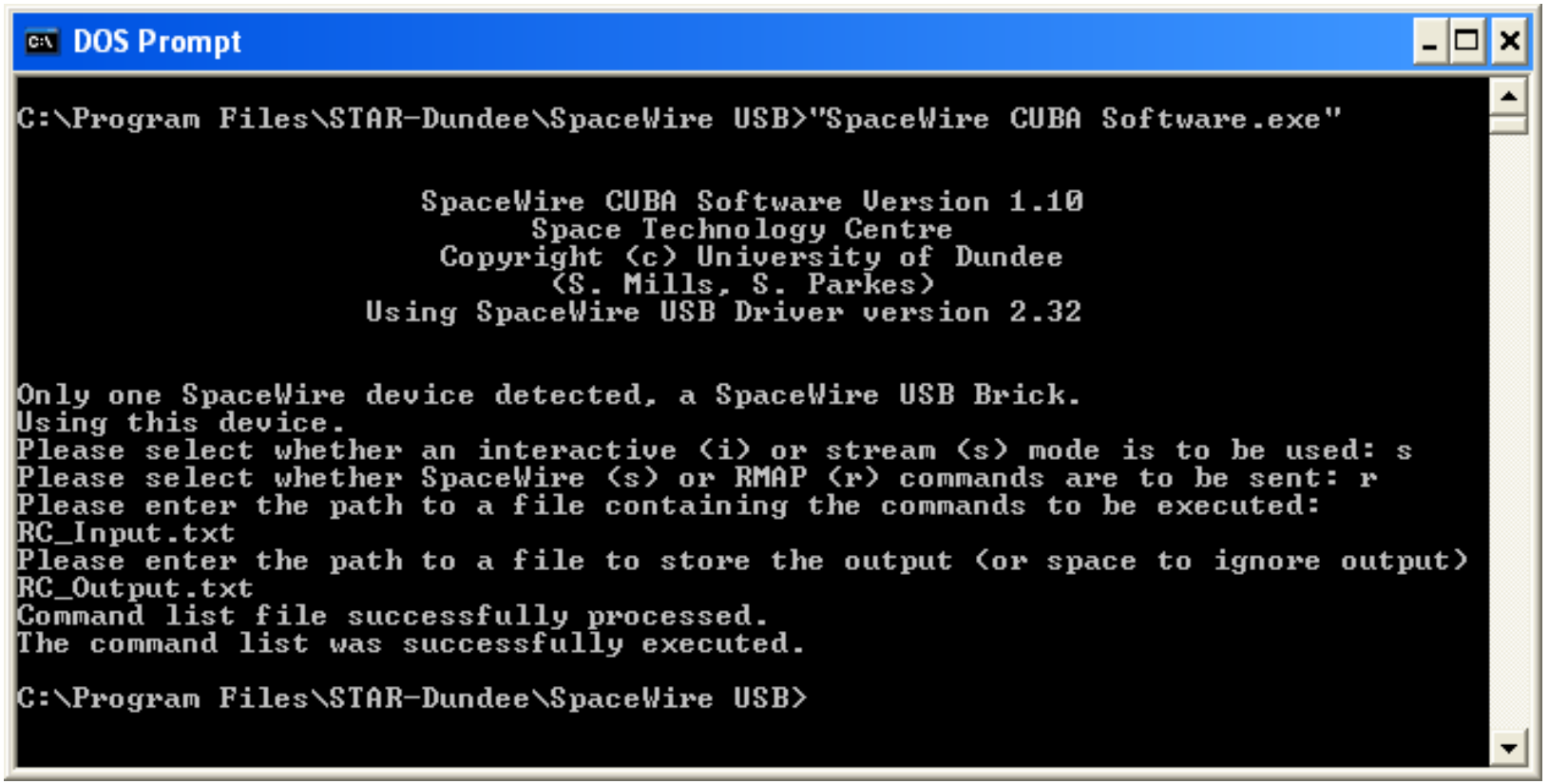
  - Example Output:

```
/* Base = Hexadecimal */
F 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff EOP
F 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff EOP
```

❑ RMAP Interactive Mode



❑ RMAP Command Stream Mode



❑ RMAP Command Stream Mode

– Example Input:

```
/* Send a read command to address 0x106 */
CommandType: r;
DestinationAddress: 0 FE;
SourceAddress: FE;
IncrementTarget: F;
DestinationKey: 20;
TransactionIdentifier: 0;
ReadWriteAddress: 106;
ExtendedReadWriteAddress: 0;
DataLength: 4;
```

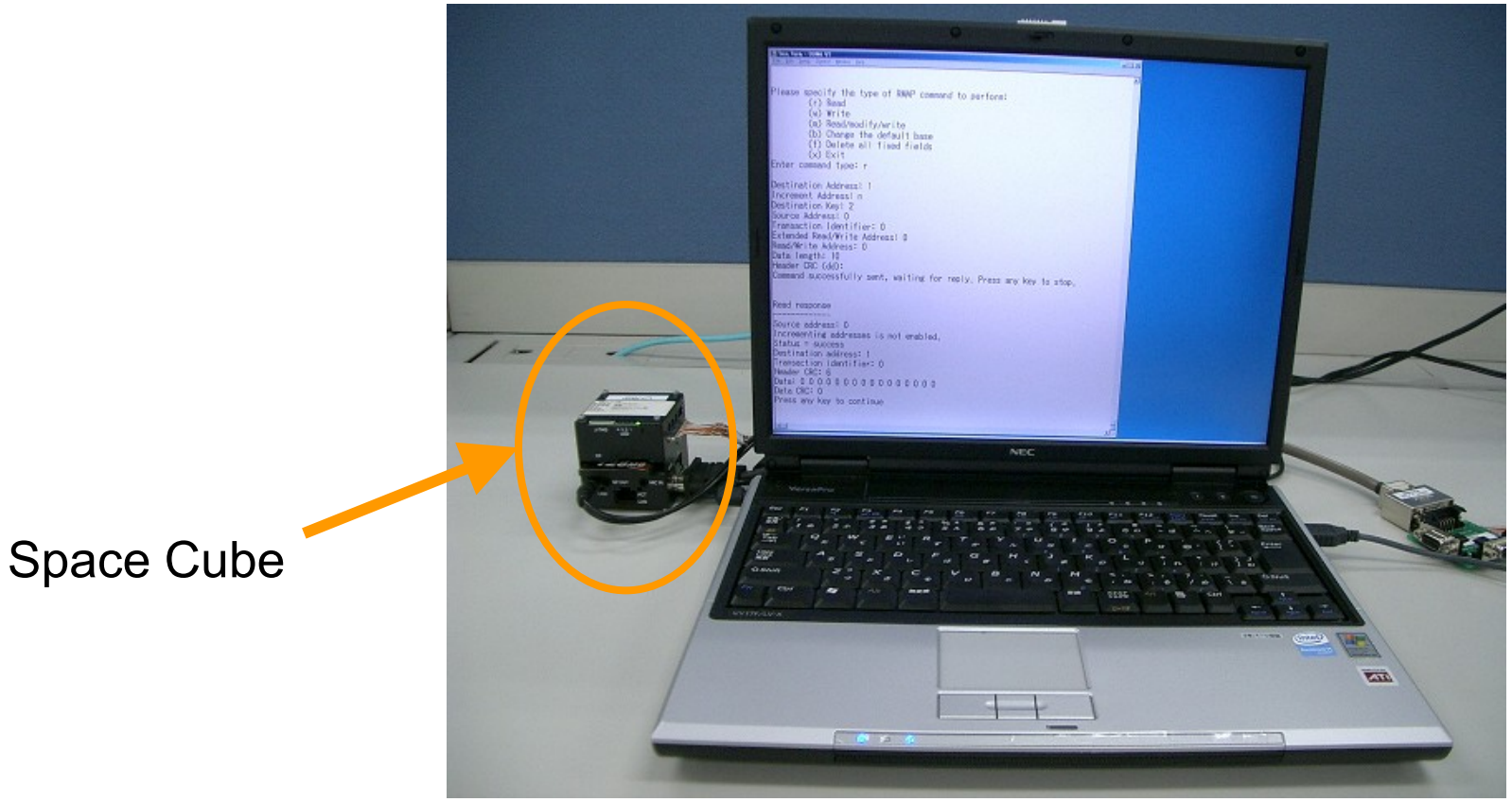
❑ RMAP Command Stream Mode

– Example Output:

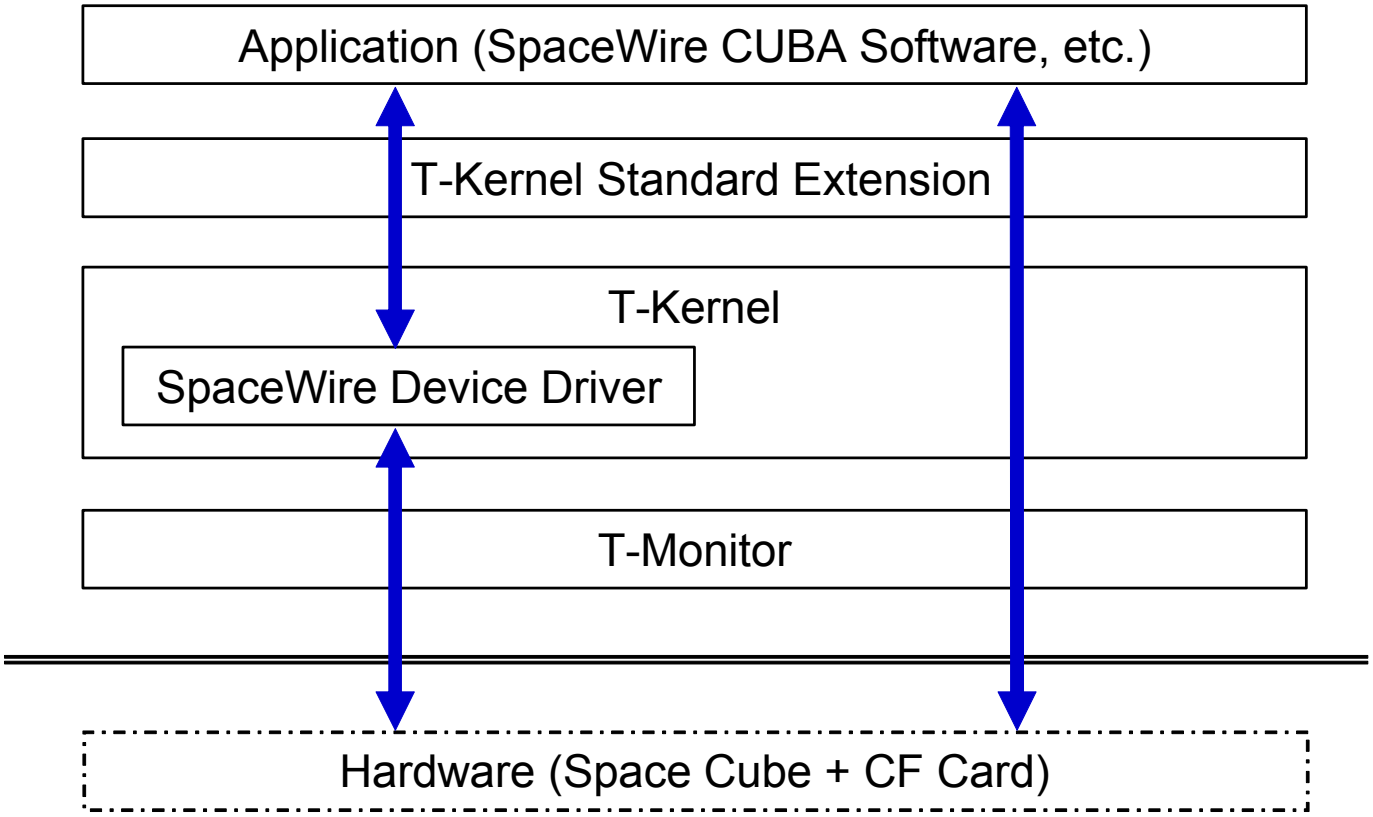
Read response

Source address: 376
Incrementing addresses is not enabled.
Status = success
Destination address: 376
Transaction identifier: 0
Header CRC: 362
Data: 0 0 0 0
Data CRC: 0

❑ SpaceWire CUBA Software on Space Cube



❑ Common API on T-Kernel and Windows



Summary

- **Joint collaboration among European and Japanese SpaceWire community**
  - Used as a **reference** in a development process.
    - (ex) Joint project like Bepi-Colombo/MMO
  - Reflects various point of view of members participated in the development project.
    - Detail understanding of SpaceWire and RMAP specification including off-nominal state has clarified.
    - The concerns have already reflected on the latest RMAP specification
    - Community members have already enjoyed the benefit of SpaceWire CUBA Software in a sense.
- **SpaceWire CUBA Software is distributed from STAR-Dundee, Ltd. for SpaceWire user community without any restriction.**