Space-Wire Applications for the MMO Spacecraft in BepiColombo Mission

T.Takashima, H.Hayakawa, H.Ogawa
ISAS/JAXA

Y.Kasaba
Tohoku Univ.

K.Masukawa, M.Koyama, M.Kawasaki, S.Ishii, Y.Kuroda
MHI

BepiColombo MMO project Data handling team
NTS, NEC soft, e-SOL
BepiColombo project
ESA-JAXA joint mission to Mercury

- BepiColombo will set off in 2013 on a journey lasting approximately 6 years.
- It arrives at Mercury in August 2019.
- It gathers the data during 1 year nominal operation & 1-year extension.

1. Origin and evolution of a planet close to the parent star
2. Mercury as a planet: form, interior, structure, geology, composition and craters
3. Mercury's vestigial atmosphere (exosphere): composition and dynamics
4. Mercury's magnetized envelope (magnetosphere): structure and dynamics
5. Origin of Mercury's magnetic field
6. Test of Einstein's theory of general relativity
The MMO spacecraft mainly aims the first complete study of the magnetic field and the magnetosphere of this unique terrestrial-type planet.

JAXA is responsible for its development and employment on the Mercury orbit.

The main objectives of the MMO spacecraft are as follows:

• **Structure and origin of Herman magnetic field**
  For the first comparative study of other planetary magnetic field.

• **Structure, dynamics, and physical processes of Herman magnetosphere**
  For the first complete study of other planetary magnetospheres.

• **Structure, variation, and origin of Herman exosphere**
  For the thin 'atmosphere': their generation / disappearance processes.

• **Environment of inner solar system**
  For the powerful environment near the sun and their energy processes.
### MMO Science Instruments

<table>
<thead>
<tr>
<th>Particle</th>
<th>Mass Spectrometer (MSA)</th>
<th>Low-energy ions</th>
<th>10eV ~ 30keV, dt=2sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solar Wind Analyzer (SWA)</td>
<td>Solar wind ions</td>
<td>10eV ~ 30keV, dt=4sec</td>
<td></td>
</tr>
<tr>
<td>High-Energy Electrons (HEP-e)</td>
<td>High-energy electrons</td>
<td>30keV ~ 700keV</td>
<td></td>
</tr>
<tr>
<td>High-Energy Ions (HEP-i)</td>
<td>High-energy ions</td>
<td>30keV ~ 1MeV</td>
<td></td>
</tr>
<tr>
<td>Energetic Neutral Atmos (ENA)</td>
<td>Plasma imaging</td>
<td>100eV ~ 3keV</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field</th>
<th>Magnetometer (MGF)</th>
<th>Magnetic field</th>
<th>DC ~ 64Hz</th>
<th>[MAST:5m]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MGF-O / MGF-I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Plasma Wave Investigation (PWI)</td>
<td>Electric field, Plasma wave, Radio wave</td>
<td>DC ~ 10MHz (E)</td>
<td>[probe: 15m x 4]</td>
</tr>
<tr>
<td></td>
<td>EWO / SORBET / AM2P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WPT / MEFISTO / SC-DB / SC-LF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image</th>
<th>Mercury Imaging Camera (MSASI)</th>
<th>Na-atmosphere spectral imaging</th>
<th>FOV=8°</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Dust</th>
<th>Mercury Dust Monitor (MDM)</th>
<th>Interplanetary Dust</th>
<th>PZT: Count &amp; Velocity</th>
</tr>
</thead>
</table>

- To be developed by different institutions and different countries.
- There is a suspicion of misunderstanding of the I/F.

In order to simplification of the I/F and testing, JAXA selected SpaceWire for all TLM/CMD interfaces.
Logical connection is fully Space-Wired.

<5 PI groups>
16 Sensors with SpW from
4 domestic contractors
[MHI + Nippi / MEC / SHI]
4 European institutes
[Austria/France/Sweden]
“Mr. CPU” hopes to be free from the RMAP-publishing & receipt processes. He has other critical jobs, Onboard-Data-Reduction!

He also hopes to work slowly, because of the much high-temperature environment.

So, we decide to bring up Mr. CPU indulgently. His ‘RMAP works’ are pushed to the PI-I/F FPGA.
Mission-I/F FPGA
~ The manager for SpW and Data Storage ~

- DMC
- DMC-IF FPGA
- CPU
- SUN PULSE
  - SUN PULSE Distribution
  - TimeCode Req.
- Access Control Block
  - Local-BUS I/F block
    - Bus Ctrl. (Target)
    - Address Decode
  - IRQ Ctrl.
  - PCI-bus I/F (Target) Block
- MISSION I/F x 8
  - MISSION I/F block [0]
    - RMAP Tx
    - RMAP Rx
  - MISSION I/F block [1]
  - MISSION I/F block [2]
  - MISSION I/F block [3]
  - MISSION I/F block [4]
  - MISSION I/F block [5]
  - MISSION I/F block [6]
  - MISSION I/F block [7]
- SDRAM I/F Block
  - EDAC Ctrl.
- SDRAM 256MB
- Tentative DATA Storage
- Sensor[0]
- Sensor[1]
- Sensor[2]
- Sensor[3]
- Sensor[4]
- Sensor[5]
- Sensor[6]
- Sensor[7]
TLM collection: by RMAP Read
~automatic RMAP-read process by MDP-FPGA~

- Mission Data collection Controller
- PCI-bus I/F (Target) Block
- SDRAM I/F Block
- EDAC Ctrl.
- SDRAM
- CPU
- DS-Link
- Sensor[0]
- 'Publisher'
- 'Postman'
- Collection sequence Set by CPU
TLM collection: by RMAP Read
~automatic RMAP-read process by MDP-FPGA~

1. Collection sequence set by CPU

2. RMAP Read

Mission Data collection Controller

Access Control Block

SDRAM I/F Block

EDAC Ctrl.

$\text{MISSION I/F block [0]}$

$\text{RMAP Tx}$

$\text{SpW Port}$

$\text{MISSION I/F block [1]}$

$\text{RMAP Rx}$

SUN PULSE Distribution

TimeCode Req.

Sensor[0]

DS-Link

CPU

PCI-bus I/F (Target) Block

RESET

CLK

CPU

SDRAM
TLM collection: by RMAP Read
~automatic RMAP-read process by MDP-FPGA~

1. Collection sequence Set by CPU
   - PCI-bus I/F (Target) Block

2. RMAP Read
   - Mission Data collection Controller

3. RMAP Read Reply
   - (Mission Data)

- 'Publisher'
- 'Postman'

- CPU
- SDRAM
- SUN PULSE Distribution
- TimeCode Req.
- MISSION I/F block [0]
- MISSION I/F block [1]
- RMAP Tx
- RMAP Rx
- SpW Port
- Sensor[0]
- DS-Link
- CLK
- RESET
- CLK
- EDAC Ctrl.
- SDRAM I/F Block
TLM collection: by RMAP Read
~automatic RMAP-read process by MDP-FPGA~

CPU only does ‘Set CMD-sequence’ & ‘Activate the sequence’. 
TLM collection: by RMAP Read

1. Cmd Set

Data Recorder (SDRAM)

1. Cmd Set

CPU

CMD Buffer

Read Cmd 1
Read Cmd 2
Read Cmd 3
Read Cmd 4
...

Payload I/F

2. Cmd Send
(periodical send by H/W)

Auto Send CTL

Sensor A

SpW Port

RMAP

HK data (TID="A")

Mission data (TID="B")

DPU

Address increment

Read Reply 1
Read Reply 2
Read Reply 3
Read Reply 4

Address increment

Read Reply 1
Read Reply 2
Read Reply 3
Read Reply 4

3. Classification by TID

4. Distribute with DMA by H/W

'Publisher'

RMAP

TID Check

Distribute
With DMA

'Topman'
Command: by RMAP Write

1. Command by RMAP Write by CPU

‘Publisher’

‘Postman’

CPU

PCI-bus I/F (Target) Block

Access Control Block

SDRAM I/F Block

EDAC Ctrl.

SDRAM

SUN PULSE Distribution

TimeCode Req.

MISSION I/F block [0]

RMAP Tx

RMAP Rx

SpW Port

MISSION I/F block [1]

DS-Link

Sensor[0]
Command: by RMAP Write

1. Command by RMAP Write by CPU

2. RMAP Write Reply (inc. error packet info)

‘Publisher’

‘Postman’

CPU

PCI-bus I/F (Target) Block

Access Control Block

MISSION I/F block [0]

MISSION I/F block [1]

SDRAM I/F Block

EDAC Ctrl.

SDRAM

TimeCode Req.

SUN PULSE Distribution

RMAP Tx

RMAP Rx

SpW Port

DS-Link

Sensor[0]

‘Publisher’

‘Postman’
Command: by RMAP Write

1. Command by RMAP Write by CPU

2. RMAP Write Reply (inc. error packet info)

3. Check error packet

'Publisher'

'Postman'

CPU

SDRAM

PCI-bus I/F (Target) Block

Access Control Block

MISSION I/F block [0]

MISSION I/F block [1]

SDRAM I/F Block

EDAC Ctrl.

SUN PULSE Distribution

TimeCode Req.

RMAP Tx

RMAP Rx

SpW Port

DS-Link

Sensor[0]
1. Command by RMAP Write by CPU

2. RMAP Write Reply (inc. error packet info)

3. Check error packet

CPU only does ‘Set CMD-sequence’ & ‘Activate the sequence’.
Timing Control

“Time tick” is used by synchronous timing pulse
[TLM timing (system I/F) / Spin phase of spacecraft (PI I/F)]

System I/F

Satellite time for MDP:
Satellite time is distributed by SpW-CMD (every 125ms)

Time distribution:
by TimeCode on SpW (every 1.953ms)

DMC

SpW I/F

Sun Sensor
Star Sensor

MDP

SpW I/F

SpW I/F for Mission Instruments

PI I/F

Satellite time for mission:
the time on SUN PULSE generated will be distributed by SpW-CMD
(just after the TimeCode distribution)

Sun Pulse (~0.25Hz):
Distributed by TimeCode on SpW

Sensor (User)

Each SI (mission)
Summary

- All over the MMO TLM/CMD communications are designed using SpaceWire I/F.

- Data collection sequence from mission sensors is controlled by the Mission I/F FPGA by RMAP Read / Read-reply (TLM) & RMAP Write / Write-reply (CMD). CPU only does the set up & start trigger.

- Time-Code on SpaceWire is used to distribute spacecraft time to MDP (system I/F) & the timing of sun-pulse (PI I/F).