

SHAPI, MAP and DMAP.

Martin Killenberg



11th June 2015

MSK Collaboration Workshow 2015, Warsaw, Poland

Standardised register set to identify hardware (PICMG recommendation)

Standard Device Registers

31	16	15	0		
Magic Word		SHAPI Version		0x00	ro
First Module Address				0x04	ro
Hardware ID		Hardware Vendor ID		0x08	ro
Device Firmware ID		Device Vendor ID		0x0C	ro
Firmware Version				0x10	ro
Firmware Timestamp				0x14	ro
Firmware Name				0x18	ro
				0x1C	ro
				0x20	ro
Device Capabilities				0x24	ro
Device Status				0x28	ro
Device Control				0x2C	rw
Device Interrupt Mask				0x30	rw
Device Interrupt Flag				0x34	ro
Device Interrupt Active				0x38	ro
Scratch Register				0x3C	rw

Standard Module Registers

31	16	15	0		
Magic Word		SHAPI Version		0x00	ro
Next Module Address				0x04	ro
Module Firmware ID		Module Vendor ID		0x08	ro
Module Version				0x0C	ro
Module Name				0x10	ro
				0x14	ro
Module Capabilities				0x18	ro
Module Status				0x1C	ro
Module Control				0x20	rw
Interrupt ID				0x24	ro
Interrupt Flag Clear				0x28	rw
Module Interrupt Mask				0x2C	rw
Module Interrupt Flag				0x30	ro
Module Interrupt Active				0x34	ro

- Standardised by PICMG
- Identify hardware without knowing it
 - Automatic driver selection
 - Automatic MAP file name generation
- Use known parts of unknown hardware / firmware

Discussion: Automate Map file access

- Naming scheme
 - Name
 - ID
 - Revision
- Where to put the files?
 - Network drive
 - Data base

Synchronise with XFEL GmbH

Plain text mapping files

# module.name	n_elements	address	size	bar	width	fracbits	signed
BOARD.WORD_FIRMWARE	1	0x00000000	4	0	32	0	0
BOARD.WORD_COMPILATION	1	0x00000004	4	0	32	0	0
BOARD.WORD_STATUS	1	0x00000008	4	0	32	0	0
BOARD.WORD_USER	1	0x0000000C	4	0	12	3	1
ADC.WORD_CLK_CNT	2	0x00000010	8	1	32	0	0

New features

feature	BSP	RapidX	MTCA4U
module	✓	✓	✓
frac bits	✓	—	✓
user info	requested		

XML Mapping

- Merge Łukasz's proposal and the XFEL implementation to SHAPI XML
- Synchronise MSK and XFEL
- Make a proposal to PICMG?

```
# Name  device_node      map_file
SIS8300 /dev/pcieunis6  llrf.map
```

Implicit assumption

- All devices are PCIe

Currently being implemented in MTCA4U

- TMCB via Ethernet

How to extend the dmap file?

PICMG Standard Device Model: Identify device via URI

```
//<Host Identifier>/<Interface>[:<Instance>][;<Protocol>][=<Parameters>]]
```

Proposal for dmap files:

```
# Name  SDM-URI                                     (map_file)
SIS8300 sdm:///./pcie:pcieunis6                llrf.map
TMCB    sdm:///./socket:mytmcdb.desy.de;tmcdb  tmcdb.map
```