» User Guide «

IPMI Firmware User Guide

for the

AM5901/AM4901

MCH Modules

kontron

If it's embedded, it's Kontron.

Revision History

Publication Title:	IPMI Firmware User Guide for the AM5901/AM4901 MCH Modules	
Doc. ID:	1035-8760	

Rev.	Brief Description of Changes	Date of Issue
1.0	Initial issue	8-Dec-2009
2.0	Updated to include software changes for R29	16-Jun-2010
3.0	Updated to include software changes for R30/R31	20-Oct-2010

Imprint

Kontron Modular Computers GmbH may be contacted via the following:

MAILING ADDRESS

Kontron Modular Computers GmbH

Sudetenstraße 7

D - 87600 Kaufbeuren Germany

For information about other Kontron products, please visit our Internet web site:

TELEPHONE AND E-MAIL

+49 (0) 800-SALESKONTRON

sales@kontron.com

www.kontron.com

Disclaimer

Table of Contents

List of Tables

This page has been intentionally left blank.

1. Introduction

1.1 Terminology and Acronym Definitions

The following table provides descriptions for terms and acronyms used in this guide. The descriptions are derived primarily from the IPMI specifications.

Table 1:   **Terminology and Acronym Definitions**

TERM or ACRONYM	DESCRIPTION
Carrier Manager	A logical function that manages and controls the AdvancedMC Modules, OEM Modules, PMs, and CUs in a MicroTCA Carrier through a set of signals controlled by the PMs together with the IPMB links
Cooling Unit (CU)	A subassembly including fans or blowers to move air to cool a MicroTCA Shelf and related support electronics
Electronic Keying or E-Keying	Abbreviation for Electronic Keying. Electronic Keying defines the process in which a MicroTCA Shelf determines if the Control and Fabric interfaces on a Module are compatible with the MicroTCA Shelf interconnects and the other Modules they reach.
EMMC	Enhanced Module Management Controller, used on Cooling Units, Power Module, and OEM Modules. See Module Management Controller (MMC).
Fabric Interface	The set of MCH Fabric Channel interfaces that provides up to seven Fabric Channels to the AdvancedMCs
Fabric, Fabric Channel	The connection where the MCH is the endpoint of a Slot-to-Slot connection of one-to-four ports on an AdvancedMC
FRU	Field Replaceable Unit, any entity that can be replaced by a user in the field
FRU Device ID	A value that uniquely identifies a FRU relative to an IPM Controller. In MicroTCA the most frequent use of FRU Device ID is to uniquely identify a FRU within a MicroTCA Carrier relative to the Carrier Manager.
FRU Information	Data that describes a FRU with an emphasis on data that characterizes the FRU. Format for this data is described in IPMI Platform Management FRU Information Storage Definition and extended herein.
Handle	An item or part used to insert or extract Modules in and out of Slots
Hot Swap	To remove a component (e.g., an AdvancedMC Module) from a system (e.g., an MicroTCA Shelf) and plug in a new one while the power is still on and the system is still operating
Hot Swap Switch	A switch that is integrated with the Module Latch Mechanism so that its state reflects the state of the Latch. The Hot Swap Switch is disconnected when the Module Handle is fully inserted.
I2C	Inter-integrated Circuit bus, a multi-master, two-wire serial bus used as the basis for IPMBs

Table 1:   **Terminology and Acronym Definitions**

TERM or ACRONYM	DESCRIPTION
Intelligent FRU	A FRU containing a management controller. Intelligent FRUs include the AdvancedMCs, MCHs, CUs, PMs, and OEM Modules, etc.
IPMB	Intelligent Platform Management Bus. The lowest level hardware management bus as described in the Intelligent Platform Management Bus Communications Protocol Specification.
IPMB-0	A dual redundant IPMB that connects MCMCs and EMMCs in a MicroTCA Carrier. Electrically and logically separate from the Local IPMB (IPMB-L).
IPMB-A, IPMB-B	Intelligent Platform Management Buses A and B, respectively. Refers to the two redundant IPMBs that aggregate into IPMB-0.
IPMB-L or Local IPMB	Connects AdvancedMC's MMC with the MCH's MCMC. Electrically and logically separate from the MCMC's IPMB-0.
IPMI	Intelligent Platform Management Interface. A specification and mechanism for providing inventory management, monitoring, logging, and control for elements of a computer system as defined in the Intelligent Platform Management Interface Specification.
LED	Light Emitting Diode
LUN	Logical unit as defined by IPMI
Managed FRU	Either an Intelligent FRU or a FRU that is represented by an Intelligent FRU via a FRU Device ID
Management Power (MP)	The 3.3 V power for a Module's management function, individually provided to each slot by the MicroTCA Shelf
MCH	See MicroTCA Carrier Hub
MCMC	See MicroTCA Carrier Management Controller
MicroTCA Carrier Hub (MCH)	An assembly providing MicroTCA Carrier functions needed to support up to twelve AdvancedMCs including MCMC, optional ShMC, optional Fabric switch, and clock.
MicroTCA Carrier Management Controller (MCMC)	Management controller on the MCH. The required management controller that interfaces to AdvancedMC MMCs via IPMB-L and to CU, PM, and OEM Module EMMCs via IPMB-0.
MMC	The required management controller on an AdvancedMC Module which interfaces to the MicroTCA Carrier Manager on the MCH via IPMB-L
Module	Refers to any MicroTCA Module types, including CU, PM, MCH, OEM Module, or AdvancedMC
Payload	The primary function that a FRU provides. This includes all the hardware on the FRU except that associated with management. It may also include the firmware, operating system and application software running on the Payload hardware.

Table 1:   Terminology and Acronym Definitions

TERM or ACRONYM	DESCRIPTION
Payload Power	The nominal 12 V supply power, individually switchable for each slot
PHY	Physical Layer Device
Power Module (PM)	Front accessible Module that controls the power entry, conversion and distribution of 12 V Payload Power to the Modules within a MicroTCA Shelf
Sensor Data Record (SDR)	A data structure record that describes the properties of a sensor, management controller or FRU Information device. Defined in IPMI.
Shelf	An electronic assembly consisting of the Subrack, Backplane, Modules, cooling devices, power subsystems, etc. Also historically known as a chassis. Shelves are usually mounted in Frames.
Slot	The union of a Connector and a Card Guide that defines the position of one AdvancedMC, MCH, Power Module, OEM Module or CU. Slots are similar in concept to the Bays used in the AMC.0 specification. A MicroTCA Subrack typically contains multiple slots.
Slot Number	Uniquely identifies a slot within a Shelf
System Event Log (SEL)	Persistent Storage for events as defined by IPMI

1.2 Related Publications

The following publications contain information relating to this product.

Table 2:   Related Publications

PRODUCT	PUBLICATION
IPMI	IPMI Specification V1.5 Rev. 1.0; V2.0 Rev. 1.0
MicroTCA	PICMG® MTCA.0 Micro Telecommunications Computing Architecture R1.0
AMC	PICMG® AMC.0, Advanced Mezzanine Card Specification R2.0 PICMG® AMC.2, Gigabit Ethernet R1.0
AM5901	AM5901 User Guide, ID: 1032-0648, Rev. 1.0
AM4901	AM4901 User Guide, ID: 1022-5813, Rev. 1.0

1.3 Features Overview

The following listing is an overview of the basic functions supported by the AM5901/AM4901 IPMI Firmware.

- Fast start

- Supports 6 AMC modules

- Firmware upgradable in the field

- Command Line Interface (CLI) with IPMI support

- CLI operable before being activated by Hot Swap switch

- Ethernet Management Port has VLAN support

- Ethernet Management Port supports IPMI Over LAN (IOL)

- Ethernet Management Port supports Serial Over LAN (SOL) for the CLI

- Dynamic Sensor Data Repository

- Non-volatile System Event Log

- Non-volatile configuration parameters

- Automatic and/or manual configuration for adaption to various environments

- AMC module's payload power can be activated permanently

- Activation, deactivation and power/current management

- Activation, deactivation and power/current parameters may be defined by the user independent of back plane FRU data

- Supports FCLKA which is implemented on the backplane

- Supports Fclock e-keying

- Manual configuration/simulation of backplane, Fclock e-keying data may be defined by the user

2.   Operation

The MCH may only be operated in systems which provide forced airflow convection cooling in accordance with the MTCA.0 specification.

Before powering up the MCH ensure that the switch for the Ethernet management port is set as required for the operation planned. Refer to chapter "5.2. Ethernet Management Port" and chapter "11.5 R31".

2.1   Start-up of the MCH after Power On

Once system power is applied, the MCH is fully operational after a short start-up time. This is independent of whether or not the MCH is activated (Hot Swap switch closed, blue LED off).

The MCH responds with the following Command Line Interface (CLI) indication:

```
Firmware 0x9000 R<n> Prod.Id <p> Manu.Id <m>
```

where: <n> = Version number 10… of the current MCH firmware

         <p> = Product ID of the MCH (e.g. 0x1325 = 4901 for AM4901)

         <m> = Manufacturer ID (0x3A98 for Kontron's IANA ID 15000)

The Command Line Interface, the Ethernet switch, and the temperature control are always in operation.

Normally communications via the IPMB-L bus are always possible between the MCH and the AMC modules as well as between AMC modules. IPMI commands can either be invoked over the CLI or via an AMC processor module.

The MCH begins operation using the configuration parameters which were set during the last operation session. These parameters may be read or modified at any time. To configure a new MCH or re-configure an MCH to a modified environment, refer to the following chapter.

2.2   Adapting to a New Environment

As soon as the Hot Swap switch is activated (closed), the MCH accesses its own FRU data (CM FRU data: FRU 0 and MCMC FRU data: FRU 3) and that of the backplane (FRU 2 or FRU 253).

The MCH FRU data serves to recognize the MCH hardware type which results in the automatic assignment of general configuration parameters (Common Options). Of importance for the recognition of the hardware type is the FRU data Board Product Name in the CM FRU data.

The backplane FRU data serves to recognize the type of backplane and the automatic assignment of the configuration parameters required by the environment (Environmental Options). Of importance for the recognition of the backplane type is the FRU data Board Part Number in the Backplane FRU data.

During the activation of the MCH, all non-readable FRU data are reported via a status indication. The following indications are possible:

```
FRU 0 (CM) DATA MISSING

FRU 2 = 253 (BACKPLANE) DATA MISSING

FRU 3 (MCMC) DATA MISSING
```

For stability reasons, default parameters are selected for all non-readable FRU data provided the current valid configuration parameters are not protected against modification.

It is always possible to manually modify configuration parameters. The status of the currently used configuration options can be determined by invoking the CLI command "showid". The response to this command will indicate if the corresponding FRU data were recognized or not.

If the MCH detects non-readable FRU data during power up, the configuration parameters will be set to default if they are not protected against modification. The MCH is then automatically reset in order to activate the "new" parameters. Before resetting occurs, one or more of the following indications will be displayed:

```
RESET FOR ADAPTION TO MCH TYPE WILL BE EXECUTED

RESET FOR ADAPTION TO BACKPLANE TYPE WILL BE EXECUTED
```

The last settings of the configuration parameters prior to power off, a manual reset via the CLI command "rs", or a Watchdog Reset are retained. Turning off power immediately after a parameter change, however, should not be done as this can cause the storing of the parameters to fail. Performing a manual reset via the CLI command "rs" ensures a safe storage operation.

The so stored parameters will survive a restart of the MCH unchanged provided that neither new FRU data is recognized nor that parameters were protected against automatic modification.

After changes in the environment (for example, the installation of the MCH in a different type of backplane) an automatic adaptation first takes place when the Hot Swap switch is closed. In addition, change protection of parameters requires consideration.

When the MCH is activated for the first time in a rack, an automatic adaption to the required environmental options parameters will be executed followed by a reset. Please refer to chapter "Environmental Options (eopts)".

If the backplane "is known" i.e. there are special parameters for it then the following CLI output will occur:

```
NOTE: NEW BACKPLANE FOUND. MANUAL FCLOCK OPTIONS WILL BE CLEARED.

PLEASE CHECK FCLOCK MODE WITH COMMAND 'fclockopts'

RESET FOR ADAPTION TO BACKPLANE TYPE WILL BE EXECUTED
```

If the backplane "is unknown" then a standard environment is assumed:

```
NOTE: BACKPLANE IS UNKNOWN. ASSUME STANDARD POWER MODULE AND
STANDARD COOLING UNIT.

DEFAULTS OF EOPTS HAVE BEEN SET.

RESET FOR ADAPTION TO BACKPLANE TYPE WILL BE EXECUTED
```

2.3 Shelf Address

The shelf address / carrier number is usually set via a DIP switch on the backplane. The address is read one time at start-up by the MCH. For this reason, the MCH must be reset or cold started after any changing of the address. Invoking the CLI command "showshaddr" will return the shelf address / carrier number. Refer to Environmental Options for further assistance.

2.4 FRU IDs

In accordance with the MicroTCA specification, the following FRU IDs are valid for the MCH. They are used as a parameter in many IPMI commands. The table below reflects the FRU IDs from the standpoint of the MCH.

Example 1:

The FRU data of the AMC module in slot 3 is selected via the FRU-ID 7 when the MCH receives an IPMI command containing the FRU-ID as a parameter (e.g. "ipmitool ... fru print 7"). When the IPMI command is addressed directly to the AMC module, then using FRU-ID 0 will provide the same result (e.g. "ipmitool ... fru print 0"). This is true regardless of the slot position of the AMC module.

Example 2:

When activating the system or individual AMC modules via the IPMI commands "Set FRU Activation" or "Set FRU Activation Policy", the FRU-IDs in the table below apply. The IPMI commands must be addressed directly to the MCH.

Table 3:   FRU-IDs

DEVICE	FRU DEVICE ID USED BY THE MCH
Carrier Manager FRU data (CM FRU data of MCH). Is MCH internal.	0
Logical Carrier Info (Backplane FRU data)	2 (same as 253)
MCMC FRU data (MCMC FRU data of MCH). Is MCH internal.	3
AMCs in slot 1 – 6	5 - 10
Logical Carrier Info (Backplane FRU data)	253

3.   System Initialization and Operation

As long as the MCH is active (Hot Swap switch closed, blue LED off) it supports all of the system units and AMC modules under its control by activation and deactivation.

3.1   Preparation

3.1.1   Supervision of AMC Modules

Only AMC modules which are recognized by the MCH can be supervised (activated, monitored, or deactivated) by the MCH. In systems which contain standard power modules, the MCH normally recognizes automatically if a slot is occupied by an AMC module or not. In this case, operator intervention is not required.

When the MCH is installed in low cost systems which do not employ standard power modules, the occupation of an AMC slot by an AMC module is not automatically recognized. Therefore it is necessary to inform the MCH during the first system initialization that an AMC slot is occupied or not, and, if occupied, that the installed AMC module is to be supervised.

Before removing a supervised AMC module, it should be indicated to the supervisor that the slot is not occupied. Then after installation of an AMC module in the empty slot, it should be indicated to the supervisor that the slot is once again occupied. Supervision of a physically not occupied slot results in unnecessary communication errors, system loading, and leads to erroneous operations.

The CLI command "slot" and the OEM IPMI command "SetSlotPresenceParameter" serve to indicate whether or not an AMC slot should be supervised. This is only required for a rack of the type OM6062.

3.1.2   Power Management and Activation Override

Sometimes during tests it is convenient to have a CPU module in the rack which is permanently powered and not affected by activation and deactivation of other modules or the MCH or even soft resets (CLI command 'rs') of the MCH. This permanently powered CPU module may be used to communicate with the IPMI system via the CPU module's KCS interface e.g. using the free tool 'ipmitool'.

To configure a module for this type of operation, perform or ensure the following. The module's Hot Swap switch must be in open state and the module must be deactivated. Power this module unconditionally by using the CLI command 'pwr' or the OEM IPMI command 'Set Permanent Power Parameter'. Example for slot 3: 'pwr on 3'. The module's payload system will be powered and start. Please note that no E-keying for that module has been performed and therefore communication with devices on other boards might fail. To end the module's permanent powered state, shut down the payload system if needed before applying the CLI command or IPMI OEM command to switch off the permanent power.

3.2   Basic Functions after Power On

When a system is powered up, all devices (the MCH and AMC modules) in the system are activated (or not) depending on the position of the their Hot Swap switches. Logically, activation of the AMC modules can only take place after the MCH itself has been activated and if the AMC modules are supervised.

3.3 Activation and Deactivation of MCH and AMC Modules

Once the MCH is activated, it automatically controls the activation and deactivation of all AMC modules under its control in accordance with the position of the AMC module Hot Swap switches. In the process, E-keying is performed in accordance with the FRU data of the involved devices: MCH, backplane, AMC modules, and power modules if installed.

If the MCH is deactivated (Hot Swap switch open, blue LED off), then all AMC modules that are currently active (blue LED off) and supervised by the MCH are deactivated. In the opposite case, if the MCH is activated (Hot Swap switch closed, blue LED on) all AMC modules currently supervised by the MCH which have their Hot Swap switches closed are also activated.

Activation or deactivation of the entire system or individual AMC modules is possible by issuing standard IPMI commands to the MCH in accordance with the applicable specification. The IPMI commands "Set FRU Activation" and "Set FRU Activation Policy" are available for this purpose. The Hot Swap switches of the devices addressed must be in the closed state.

Conflicts arising from the activation or deactivation of the MCH or an AMC module as a result of too quickly opening or closing of the device's Hot Swap switch are solved by first completing the action that was started, and then determining how to further proceed on the basis of the current position of the involved Hot Swap switch.

3.3.1 Sequence of Activation and Deactivation of AMC Modules

The sequence of activation is defined by the backplane FRU data. There the priority of the slots is defined. The sequence of deactivation is the reverse. If this sequence is not sufficient and it is not possible or desirable to adapt the FRU data to the situation (e.g. for tests only), there is the CLI command 'prioopts' and the IPMI OEM command 'Set Priority Options Parameters' for manually setting slot priority parameters.

3.3.2 Timing for Activation and Deactivation of AMC Modules

The timing during activation and deactivation is defined by the backplane FRU data. During activation the maximum wait time for a module to become operational and the wait time after the activation of every slot is defined. If the wait time specified is not sufficient and it is not possible or desirable to modify the FRU data to support the current activity (e.g. for tests only), the CLI command 'activopts' and the IPMI OEM command 'Set Activation Options Parameters' are available to manually set timing parameters.

If the maximum activation wait time for a given module is exceeded before it reaches state M4, a warning message will be displayed at the console:

```
WARNING: ACTIVATION TIMEOUT FOR MODULE IN SLOT <slot number>.
CHECK MODULE AND PARAMETERS! 'activopts' MODE IS <mode>.
```

This message indicates that the module in slot <slot number> has caused an activation timeout during the activation sequence and will be ignored. The specified activation sequence has failed.

The operator should then verify the status of the module in question and correct the selected mode and time parameters as required using the CLI command or IPMI OEM command indicated above.

3.3.3 Granting Payload Power During Activation of AMC Modules

The granting of payload power during activation of an AMC module is a function of the backplane FRU data, the AMC module FRU data, and the standard power module if present. Based on the AMC modules' current requirements, and limits defined for the system as well as individual slots, module payload power is granted or denied.

If the FRU data is not correct and it is not possible or desirable to modify the FRU data to support the current activity (e.g. for tests only), the CLI command 'currentopts' and the IPMI OEM command 'Set Current Options Parameters' are available to manually set current parameters.

During the activation of modules, all current capacity problems are reported via a status indication. The following indications are possible if backplane FRU data are not found:

```
CARRIER ACTIVATION AND CURRENT DESCRIPTOR MISSING FOR SLOT
<1..6>.

CURRENT REQUIREMENT RECORD MISSING FOR MODULE IN SLOT <1..6>.

NO CARRIER POWER POLICY RECORD FOUND.
```

The following indications are possible if backplane FRU data concerning current is available or current data is manually defined but the values being offered are not sufficient:

```
CURRENT POOL WAS EMPTY WHEN REQUIRING CURRENT FOR SLOT <1..6>.

REQUIRED CURRENT TOO HIGH FOR SLOT <1..6>.

'currentopts' MODE is <mode>.
```

Please note that the sequence of power grants or power denials may differ from the activation sequence of the modules or rather the slots.

Please note that an AMC module whose FRU data is missing or in whose FRU data the current requirement record is missing or contains the value of 0 [100mA] will never become activated.

It is possible to switch off all current restrictions being described above by switching off the current mode. CLI command 'currentopts' and the IPMI OEM command 'Set Current Options Parameters' may be used for that. Please note that this is not recommended for normal operation and may cause an overload of power modules and contacts.

3.4 Temperature Regulation

Temperature regulation is normally active as soon as the backplane's configuration parameter is automatically set. Manual activation or deactivation can also be accomplished by issuing the CLI command "eopts" or the IPMI command "Set Environmental Options Parameter".

The regulation parameters can be displayed or set by issuing the CLI command "tcp" or the OEM IPMI commands "Set Event Driven Temperature Control Parameter" and "Get Event Driven Temperature Control Parameter".

When the temperature regulation is deactivated, the fan speed can be manually set by issuing the CLI command "fan" or the OEM IPMI command "Set Fan Speed Parameter". As long as the temperature regulation is deactivated, the previously set fan speed remains in effect.

Normally the fan control foresees 16 different fan speed levels (0 to 15 = 0 to F in CLI command). There are, however, systems whose fans can only be on or off. This can lead to a very coarse regulation and it could be advisable to deactivate the fan control and let the fan(s) operate at their own fixed speed.

Please note: For OM6060 systems there is the restriction that the fan is only capable of being switched on or off. If the calculated fan speed level is <= 3 the fan will be switched off, otherwise the fan will run at full speed.

3.4.1 Regulation Parameters

The following temperature regulation parameters are available. Input and output to the command line is done via the CLI command "tcp" without "0x" or "h" affixed. The default values are in parenthesis, time is in seconds.

- initialFanLevel (8)
- minFanLevel (3)
- coolingFanIncreaseTimeout (3c = decimal 60)
- coolingFanDecreaseTimeout (3c = decimal 60)
- fanLevelStepUp (2)
- fanLevelStepDown (1)
- normalStableTime (708 = decimal 1800)

3.4.2 Regulation Algorithm

The MCH registers critical as well as non-critical temperature events within the system. There are three possible states which are registered:

- State 0 = no active temperature event
- State 1 = at least one non-critical temperature event is active
- State 2 = at least one critical temperature event is active

The MCH always starts in the state 0 and with the fan speed "initialFanLevel".

In state 0, the fan speed is then reduced by the value of "fanLevelStepDown" after "coolingFanDecreaseTimeout" seconds until the level "minFanLevel" is achieved. After further "normalStableTime" seconds, the "minFanLevel" is decreased by one level with a corresponding reduction in fan speed.

Transition from state 0 to state 1 results in the immediate increase of "minFanLevel" and the fan speed by the level "fanLevelStepUp". Following this, the fan speed will be increased by "fanLevelStepUp" every "coolingFanIncreaseTimeout" seconds until either state 0 or state 2 occurs.

By a transition to state 2, the fan speed increases immediately to the maximum possible speed and remains there until the system transitions to state 0.

3.5 E-Keying

In order to ensure proper operation of the AMC modules in the system, the topology of the employed backplane must support the required connectivity. The MCH's E-keying process requires during the activation and deactivation of AMC modules that the module's line drivers be switched electronically on and off. The switch-on takes place before each AMC module is supplied with payload power (so called "early E-keying").

The FRU data of supported AMC modules may contain a maximum of:

- 16 AMC channel descriptors
- 22 AMC link descriptors
- 4 AMC Clock descriptors

If this amount is exceeded, no ports of the affected AMC modules will be enabled.

3.5.1 FCLKA E-Keying

The MCH being described here is not able to supply the clock for PCI Express (CLK3/FCLKA). This clock must be supplied by the backplane or an AMC module.

The MCH has three different strategies to be able to work in all possible clock environments. The option "fclockMode" in the environmental options rules the MCH's behavior during Fclock e-keying.

When the MCH identifies the type of rack or backplane respectively where it is running it automatically adjusts the "fclockMode" options in the "environmental options" in case the environmental options are not yet 'locked'. Please refer to chapter "Environmental Options/Defaults" where the possible "fclockMode" modes are listed which can be set depending on the rack or backplane type.

The "fclockMode" may be set manually as well, e.g. by CLI command 'eopts' or CLI command 'fclockopts', and can be protected against undesired automatic changing by setting the 'locked' bit if required.

Examples for CLI:

- 'fclockopts spec'
- 'fclockopts rack'
- 'fclockopts manu'

Please note that a manual change of "fclockMode" should only take place when no AMC is activated.

There are three different "fclockMode" modes defined (which one is currently set is showed e.g. by the CLI command 'fclockopts'):

- 'spec': An AMC/uTCA specification compliant Fclock e-keying will be performed. This means that any Fclock being offered by the backplane is ignored. Only the "Carrier Clock Point-to-Point Connectivity Records" (refer to AMC.0 R2.0 3.9.2.2 Table 3-29) being offered by the backplane and the "Direct Clock Descriptors" (refer to AMC.0 R2.0 3.9.2.3 Table 3-38) being offered by the AMC modules will be processed. If two AMC modules become activated and their Fclocks lines are connected by the backplane and there is a source/receiver pair of clock descriptors then their clock ports will be enabled.

Example: OM6040D has no Fclock but slot to slot connections for it. The Fclock must be provided by an AMC.

- 'rack': It is known that a backplane: a) offers an Fclock source permanently; or b) there is a non-standard method to switch on and off the Fclock source for any slot. For both cases a) and b) no Fclock e-keying will be performed at all. I.e. the modules will not receive any clock control commands. For case b) the slot's Fclock will be switched on/off if a PCIe connection between two AMC modules will be activated/de-activated. Note: Kontron Processor AMC boards need no support for that. Either they are always able to be clock receiver or they are initialized as a clock receiver when PCIe becomes active during port e-keying.

  Example for case a): OM6062, OM6063 or compatible systems have a permanent Fclock.

  Example for case b): OM6060 or compatible systems have a switchable Fclock.

- 'manu': If 'spec' mode or 'rack' mode is not possible or not satisfying the manual mode can be selected. Use the CLI command 'fclockopts manu' to invoke that mode. Normally additional parameters have to be set before. Please refer to the description of this CLI command and the special chapter in the following.

3.5.2 Manual Fclock Mode

The fclockMode "manu" should permit operation with any desired PCIe module configuration on a backplane where at least the required physical connections for Fclock and PCIe lines between one or more pairs of slots are implemented.

These low cost MCHs provide neither an Fclock generator nor a PCIe switch. However, under certain circumstances depending on the backplane configuration, it may be possible to overcome this situation and achieve satisfactory operation. For further assistance, contact Kontron.

Another problematic situation occurs if an AMC module is not able to process the 'Set Clock State' IPMI command. This command is mandatory during a spec conformant PCIe e-keying to set an AMC's fclock to source or receiver. If an AMC module does not process this command the e-keying will not be finished successfully. The problem may be solved by choosing the manual ('manu') fclock mode and set the AMC module's 'fclock strategy' to 'Do not influence the AMC's Fclock behavior'. In that case the AMC module is supposed to be a clock receiver. Please note that for the partner AMC the behavior has to be set as well. Please refer to the fclockopts parameters being described below.

To solve these and similar problems if e.g. backplane FRU data for the Fclock are missing or wrong or it is desired to force the behavior of modules during the Fclock e-keying the manual mode can be used. The following describes what is possible.

For every slot ('source slot') the following may be defined:

- an Fclock strategy
- a remote slot

The CLI command to be used is:

```
fclockopts manu <source slot> <Fclock strategy> <remote slot>
```

where:

`<source slot>`   denotes the slot for which the following parameters have to be set. Please note that 'source slot' does not mean that here resides a Fclock source.

`<fclock strategy>` defines the 'source slot's' assumed or desired Fclock behavior.

`<remote slot>`   denotes the slot which's Fclock <line is assumed to be physically connected to the <source slot>. 0 means that there no change of the point-to-point fclock connectivity data being defined for the back plane is needed for this 'source slot'. If at least for one 'source slot' the 'remote slot' is not 0 then the backplane's "Carrier Clock Point-to-Point Connectivity Records" data during Fclock e-keying will be logically replaced by the manually defined connections between 'source slots' and 'remote slots'. If all 'remote slots' are 0 then the backplane's "Carrier Clock Point-to-Point Connectivity Records" data will be used during Fclock e-keying. Please note that every 'source slot' must have a unique 'remote slot' if not 0.

If 'remote slot' is 0 then 'fclock strategy' denotes how the AMC in this slot shall be unconditionally set up like this:

  0 = Do not influence the AMC's Fclock behavior.

 1x = The AMC's Fclock shall be receiver as defined in its xth match 0 descriptor.

 2x = The AMC's Fclock shall be source as defined in its xth match 1 descriptor.

If 'remote slot' is 1… then 'fclock strategy' denotes the Fclock e-keying strategy of the MCH like this:

  1 = Use the Fclock parameters being specified in the FRU data of the AMC in   'source slot'. Use the matching ones when connecting slots.

  2 = Simulate that the AMC in 'source slot's' is a clock receiver. For this the 'remote slot' AMC's first Fclock FRU data will be copied and set to receiver to force the AMC in 'remote slot' to become a Fclock source. The 'source slot' AMC's Fclock will not be touched. Please note that for that the AMC module in the 'remote slot' must have fclock FRU data.

  3 = Simulate that the AMC in 'source slot's' is a clock source. For this the first 'remote slot' AMC's first Fclock FRU data will be copied and set to source to force the AMC in 'remote slot' to become a Fclock receiver. The 'source slot' AMC's Fclock will not be touched. Please note that for that the AMC module in the 'remote slot' must have fclock FRU data.

3.6   Watchdog

The MCH's processor provides an autonomous and continuously active Watchdog function. If the MCH Firmware does not regularly trigger the Watchdog as a result of an internal malfunction, the Watchdog forces automatically a system reset thus preventing potentially dangerous operations. This reset is an "unexpected" reset meaning it causes a complete new start of the MCH comparable to a power on reset. As a result of this reset, all AMC modules are powered down after which the system is powered up again according to the position of the Hot Swap switches.

3.7   Anomalous Behavior and Status Indications

System operational states, anomalies, and error recording are indicated by the use of LEDs, an Eventlog, and error variables.

Status information is provided by the following:

- •   IPMI System Eventlog (SEL) - by reading the System Eventlogs states such as overtemperature or under- overvoltage conditions can be detected

- •   The error variable - it can be read out via the CLI command "showerror"

System failures are indicated by the Health LED on the MCH being lit (red).

In the following cases the MCH itself will attempt to resolve critical situations:

- •   Temperature problems in the system: the fan regulation will attempt to resolve the problem

- •   Configuration problems: it is possible that a reset will be performed in order to have automatically set configuration parameters take effect

For failures, the MCH Health LED will light up red. In such cases, it is recommended to check the error variable as described above. Random runtime failures such as a communications failure (M state 7) for an AMC module are non-self-clearing. Non-self-clearing means that when such a failure is resolved it is not automatically cleared in the error variable. Such failures can only be cleared by issuing the CLI command "clearerror". Other failures are self-clearing. For example, a temperature failure clears itself when the temperature returns to normal. The clearing of a failure can lead to the Health LED going green when no other non self-clearing failure exists.

When a failure in an AMC module occurs, the Health LED of the failed module lights up red. In addition, the failure cause is usually recorded in the Eventlog of the MCH, because the AMC module has reported the failure to the MCH. Only in the event of a temperature failure, the red Health LED of the AMC module and the red Health LED of the MCH will be active.

3.8   LEDs

3.8.1   Blue Hot Swap LED

The Hot Swap LED indicates the Hot Swap state of the MCH in the case that power is applied. At power on, the LED lights up and the MCH is at first inactive. The Hot Swap switch must be closed to manually activate the MCH. To manually deactivate the MCH, the Hot Swap switch must be opened. During the state change, the LED blinks and in the active state of the MCH it is off. The MCH may only be removed from the system when it is deactivated.

3.8.2   Out Of Service (OOS) LED

The Out Of Service LED is lit as long as the MCH is not operating. This is the case when it is in an un-programmed state, during updating of the Firmware, and when resetting.

3.8.3   Health LED

The Health LED of the MCH indicates its operational state. As long as the MCH is operating, the LED blinks green ("Heart Beat"). In case of a failure, the LED blinks red, green, red, ...

3.9 Eventlog

The MCH provides the IPMI System Eventlog (SEL) with a non-volatile buffer for up to 1023 entries. In order to prevent a buffer overflow and resulting loss of event records, the buffer must be periodically emptied. Access to the Eventlog buffer of the MCH can be effected best from an AMC processor module over the IPMB-L bus using the free software tool "ipmitool". A buffer overflow is of no consequence for the MCH, but can, however, be of relevance to system management software. The events stored in the Eventlog are state changes of sensors implemented in the MCH. Possible events are described later in this guide.

3.10 Error Variables, Error Indication, and Error Clearing

The MCH provides an error variable which can be read via the CLI command "showerror". The clearing of particular bits can be effected using the CLI command "clearerror". Some errors are self-clearing when the condition that caused them no longer exists. Self-clearing and clearing per command are explicitly indicated in the following tables.

Table 4: Error Variable Structure

BITS	OBJECT	FORMAT	DESCRIPTION
31	---	---	Reserved
30	eventProcessing	bool	1 = Error when storing an event in the event log. Possibly the event log is full. 0 = No error
29	powerMissing	bool	1 = Not enough current deliverable by the power module or more current for a slot required than allowed. Refer to 1) CLI commands 'currentopts', 2) backplane FRU data 'MicroTCA Carrier Power Policy record' and 'Carrier Activation and Power Management record' and 3) OEM IPMI commands 'Set Current Options Parameters', 'Get Current Options Parameters' 0 = No error
28	fru3DataMissing	bool	1 = FRU data for FRU ID 3 (MCMC) not readable. Load the FRU data and reset the MCH. 0 = No error
27	Fru2DataMissing	bool	1 = FRU data for FRU ID 2 or 253 (Backplane) is not readable. Load the FRU data and reset the MCH. 0 = No error
26	Fru0DataMissing	bool	1 = FRU data for FRU ID 0 (CM) not readable. Load the FRU data and reset the MCH. 0 = No error
25:23	shAddrDevice	device	State of the shelf address read device. Runtime errors can be cleared.
22:20	tempCtrlDevice	device	State of the temperature control device. Please note: Does not report temperature errors. Runtime errors can be cleared.

Table 4:   Error Variable Structure

BITS	OBJECT	FORMAT	DESCRIPTION
19	badBoardHealth	bool	1 = Error reported by the MCH's Health sensor which reports various error states of MCH sensors; e.g. temperature or voltage errors. Is self clearing. 0 = No error
18	commLost	bool	1 = M state 7 for at least one AMC module detected. Can be cleared. 0 = No M 7 state
17:15	switchDevice	device	State of the Ethernet Switch device. Runtime errors can be cleared.
14:12	phyDevice	device	State of the PHY control device. Runtime errors can be cleared.
11:9	ps1EnDevice	device	State of the PS1 (AMC detection) device. Runtime errors can be cleared.
8:6	clkDevice	device	State of the FCLK_A control device. Runtime errors can be cleared.
5:3	pwrDevice	device	State of the power control device. Runtime errors can be cleared.
2:0	fanDevice	device	State of the fan control device. Runtime errors can be cleared.

Table 5: Error Formats

VALUE OF STATES	DESCRIPTION OF STATES
FORMAT DESCRIPTION (for "device" in Table 4)	
4	Runtime error: May be cleared by CLI command 'clearerror' (s. 5.2)
3	Error during MCH initialization. Not clearable.
2	ID in common options or environmental option for this device is illegal for the present configiration. Not clearable.
1	ID in common options or environmental option for this device is undefined. Not clearable.
0	No error
FORMAT DESCRIPTION (for "bool" in Table 4)	
1	Error: Not clearable.
0	No error

4. System Management

The following interfaces are available to support system management:

- Rack's power switch
- Hot Swap switch on the MCH
- CLI interface of the McH
- MCH Ethernet management port
- Processor AMCs KCS interfaces
- AMC module Hot Swap switches

Usually the power switch and/or the Hot Swap switches on the MCH and AMC modules is sufficient to activate or deactivate a previously configured system. This means that once a system is configured, no further operating elements are required. In the extreme case, the rack's power can simply be switched on or off.

The configuration of the system can be accomplished for example via the CLI interface of the MCH. Using the serial interface on the front panel of the MCH, a service terminal with keyboard or a computer can be connected. When connected to the serial interface of a computer, the computer must be operated either as a terminal with keyboard or it must provide a script program (previously created by the user) that controls the commanding operation. It is recommended to use the CLI interface when performing the initial configuration of a system. For systems which do not provide automatic slot presence detection (for example, the OM6062), a minimal initial configuration must include "slot on xx" for used slots.

Another way to system configuration is to use standard and OEM IPMI commands. All of the MCH's CLI commands for configuration have their correspondent OEM IPMI commands. Therefore instead of using the CLI commands there may be used IPMI commands to administrate the system. There are the following interfaces which might be used for sending IPMI commands to the MCH:

- The MCH's Ethernet management port
- A standard IPMI interface (KCS interface) on a processor AMC's payload

The configuration of the MCH for a certain rack type as a rule is done automatically when activating the MCH in the rack for the first time. If additional modifications are required after that, they can be performed using CLI commands. All configuration modifications may also be done by issuing OEM IPMI commands using a tool for system management support. Usually the open Linux tool "ipmitool" is used as a small management tool to send standard or OEM IPMI commands directly to the MCH. For complex management tasks such as automatic power-down or reading out of the System Event Log, "ipmitool" offers specific commands which combine several standard IPMI commands.

When a µTCA system once is running it is even possible to administrate it by sending IPMI commands from a hosted processor AMC module to the MCH via the IPMB-L bus on the backplane. For this the used communications path would then be bi-directional as follows:

Management program in AMC module (e.g. "ipmitool")

|

IPMI driver on the payload of the AMC module

|

KCS interface of the AMC module

|

MMC of the AMC module

|

IPMB-L on the backplane

|

MCMC of the MCH

|

Carrier manager of the MCH
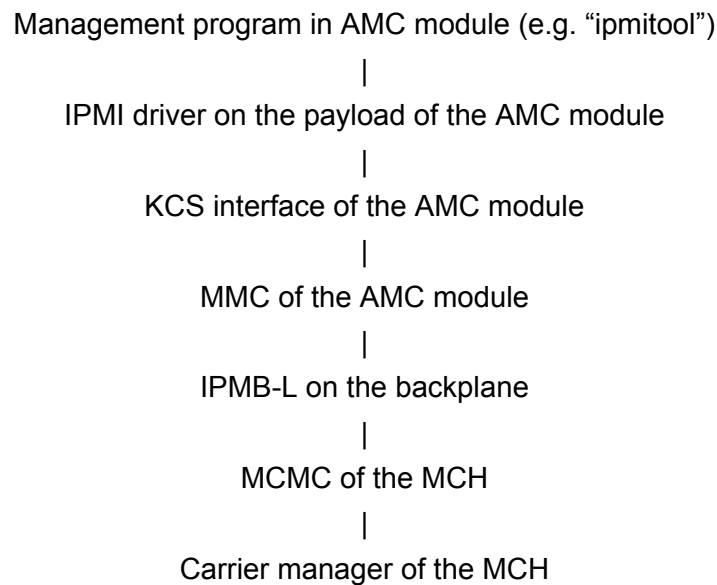
The operating system Linux® can be so configured that the IPMI driver starts automatically and that "ipmitool" can be directly called. It must also be considered that the management program is only available as long as the AMC module on whose payload it is running is activated by the MCH. It is possible to force permanent operation of the AMC module (leaving out the MCH) by issuing the CLI command "pwr" or the OEM IPMI command "SetPermPwrParameter" for the corresponding system slot for which the payload power is to remain on permanently.

5. User Interfaces on the MCH Front Panel

The MCH has two user interfaces at its front panel, the serial management interface and the Ethernet management interface. The serial interface gives access to the command line interface (CLI) where the operator can manage the rack. The Ethernet management interfaces gives access to the IPMI system. This kind of communication is called IPMI Over LAN (IOL). A combination of CLI and IOL is implemented as well, being named Serial Over LAN (SOL). These interfaces will be discussed in the following.

5.1 Command Line Interface (CLI)

The command line interface (CLI) is accessed via the serial interface on the front panel of the MCH. The CLI is not necessary for normal operation of the MCH because the only relevant operating elements available are the Hot Swap switches of the MCH and the AMC modules. The CLI is then necessary for manual configuration when the automatic configuration requires modification for example for tests or failure analysis.

5.1.1 Interface Parameters

The serial interface should be configured as follows: 9600, 8, N, 1

5.1.2 CLI Command List

In the following command listing, all command parameters are hexadecimal coded during input and output unless otherwise noted. No '0x' prefix or 'h' suffix is allowed.

Table 6:   CLI Command List

Command	Parameters	Description	User or Root
?		Same as help	UR

Table 6:   **CLI Command List**

Command	Parameters	Description	User or Root
activopts		Shows which activation and deactivation mode is currently selected (auto or manu or off). When manu mode is selected the manual parameters are displayed as well.	R
	auto|manu|off	Sets the activation and deactivation management mode for selection of the delays being used during the activation and deactivation of the AMC modules. auto: Follows the data in the FRU record "MicroTCA Carrier Activation and Power Management record". manu: Use delays as being defined in the manu parameters below. off: No activation and deactivation delays will be used.	
	manu <max. allowed ready time> <delay after slot 1> … <delay after slot 6>	Sets the manual activation mode and the parameter values for manual mode. "max. allowed ready time" defines the maximum time (hex [1s]) that will be waited until a module is active before activating the next module. "delay after slot x" defines which time (hex [100ms]) will be waited after the activation or deactivation of a module has ended before the next module will be processed.	
clearerror		Clears the runtime errors in the error variable and shows the present state of the error variable using command 'showerror'	R
cmds		Print list of commands	UR
copts		Display the current common options as one hexadecimal number and as a sequence of hexadecimal numbers	R
	<common options (hex)>	Set all common options as one hexadecimal number	R
	<option designator> <common option (hex)>	Set one common option. For the related option designator refer to the result of command 'copts'. Only the first four characters of the designators being listed are used as option designators. Note: For 'supprAM4901ErrorBeforeBoardRevR12' use 'supP' instead of 'supp'. For the option values refer to chapter 'common options'.	R

Table 6:   **CLI Command List**

Command	Parameters	Description	User or Root
current-opts		Shows which current mode is currently selected (auto or manu or off). When manu mode is selected the manual parameters are displayed as well.	R
	auto|manu|off	Sets the current mode for selecting the current parameters: auto: Follows the data in the FRU record "MicroTCA Carrier Power Policy record". manu: Use current values as being defined in the manu parameters below. off: No current limitations used.	
	manu <maxCurrentOverride]> <maxCurrent slot 1> ... <maxCurrent slot 6>	"maxCurrentOverride" defines the maximum current (hex [100mA]) that the rack's power module is able to deliver. "maxCurrent slot x" defines the maximum current (hex [100mA]) that is allowed to be consumed by a module in slot x.	
eopts		Display the current environmental options as one hexadecimal number and as a sequence of hexadecimal numbers	R
	<environmental options (hex)>	Set all environmental options as one hexadecimal number	R
	<option designator> <environmental option (hex)>	Set one environmental option. For the related option designator refer to the result of command 'eopts'. Only the first four characters of the designators being listed are used as option designators. For the option values refer to chapter 'environmental options'.	R
fclock		Display the current states of the slot's f-clock signals. States may be 'off' or 'on' or 'ssc'. Note: on or ssc means that during E-keying the slot was detected as a consumer of the rack's clock. The clock type is non ssc (on) or ssc. This command is only supported in racks which have a switchable clock generator.	R
	< slot 1 f-clock state> ... < slot 6 f-clock state>	This command is only for tests: Set the state of the f-clock to 'off' or 'on' or 'ssc'.	R

Table 6:   CLI Command List

Command	Parameters	Description	User or Root
fclock-opts		Show currently active FCLKA options	R
	spec	Select AMC/µTCA specification compliant Fclock e-keying	R
	rack	Select Fclock generated by the backplane. Fclock source for the slot is always on or switched on by the MCH when PCIe is active. No clock e-keying for AMC modules.	R
	manu	Select AMC slot specific manual selection of parameters	R
	manu <source slot number> <source slot's Fclock strategy> <remote slot number>	Enter for one slot with number <Source slot number > (1..6) the manual parameters <Source Slot's Fclock strategy> (refer to below) and <Remote slot number> (0..6). **Usage of parameter 'remote slot number':** Only if for at least one source slot a 'remote slot number' > 0 is defined the backplane's clock ptp FRU data will be replaced for ALL slots by FRU data being defined by all pairs of 'source slot number' and 'remote slot number' parameters. Remote slot number 0 means no remote slot connected. Each used remote slot number (except 0) must be unique, i.e. each source slot may only be connected to one remote slot. **Usage of parameter 'Fclock strategy':** 'Fclock strategy' if 'remote slot number' is defined (1..) = 1 = AMC Fclock parameters as specified in FRU data Use matching ones 2 = AMC Fclock shall be receiver Copy/modify remote AMC's parameters 3 = AMC Fclock shall be source Copy/modify remote AMC's parameters. 'Fclock strategy' if no 'remote slot number' is defined (0) = (AMC shall be source/receiver/fixed, e.g. by switch or default state) 0 = Do not touch AMC's Fclock Use AMC's fixed or default state." 1x = AMC Fclock shall be receiver as defined in xth match 0 descriptor 2x = AMC Fclock shall be source as defined in xth match 1 descriptor	R

Table 6:   **CLI Command List**

Command	Parameters	Description	User or Root
fan	<speed (hex)>	Set the fan speed level to 0…f (min…max). Please note that the manual setting is only possible if the temperature control is switched off	R
		Display the real fan speed [rpm]	R
help		Print online help of all commands with parameter descriptions	UR
lan	print 1	Display all current LAN parameters for the management port	R
	set 1 macaddr <x:x:x:x:x:x>	Set the management port's MAC address (x = hex)	R
	set 1 ipaddr <x.x.x.x>	Set the management port's IP address (x = dec)	R
	set 1 netmask <x.x.x.x>	Set the management port's subnet mask (x = dec)	R
	set 1 vlan <x>	Set the vlan ID x (x = dec, 0 < x < 4095)	R
	set 1 vlan off	Set the vlan support off	R
	set 1 access on	Set the management port channel access mode to on	R
	set 1 access off	Set the management port channel access mode to off	R
led		Access to Carrier Management LEDs	
	on|off	MCMC (on) or Switch (off) controls CMLEDs 1..13	R
	set|clr <led#>	Set or clear a single LED (requires 'led on'). <led#>: 1 - d = LED number (hex)	R

	all <mask>	Set or clear all LEDs (requires 'led on').
		<mask>: 0 - 2ffe One bit per CMLED 1..13
		        0 All LEDs off
		      2ffe All LEDs on
			R

login	<current password>	Log in using the current password to be able to use root (R) commands. If no password was defined until now no login is needed.	UR
logout		From now on no root (R) commands can be used if a password is currently defined.	R
md	[<adr>]	Display memory contents. <adr> in hex to define start of display; default is end of last display	UR

Table 6:   CLI Command List

Command	Parameters	Description		User or Root
mb	<adr> [<item-to-write>]	Display/change single memory locations of type	Byte (8 bit)	R
mw			Word (16 bit)	
ml			Long (32 bit)	
passwd	<new password >	Set a new password as sequence of 5...8 printable ASCII characters. After having set a password you must log in (command 'login') before you can use a root only (R) command. If the <new password> is 'killpass' then no current password is defined and no login is needed to use root (R) commands.		R
prioopts		Shows which priority mode is currently selected (auto or manu or off). When manu mode is selected the manual parameters are displayed as well.		R
	auto|manu|off	Sets the priority mode for activation and deactivation of the AMC modules. auto: Follows the data in the FRU record "MicroTCA Carrier Activation and Power Management record". manu: Use priorities as being defined in the manu parameters below. off: No priority used. Random sequence is used during activation and deactivation.		
	manu <priority slot 1 (hex)>. <priority slot 2 (hex)> ... <priority slot 6 (hex)>	Sets the manual priority mode and the parameter values for manual mode. Note: Use the priority values to define the sequence of activation and deactivation. Slot with priority value 1 will be activated first and deactivated last. If Bit 7 of the priority value is set additionally (e.g. 81h for priority 1) means that this slot will be activated and deactivated randomly. Example: 1 3 2 4 5 86 activates slot 1 then slot 3 then slot 2 etc. The deactivation is vice versa. Slot 6 will become activated and deactivated randomly. The priority setting feature is needed when e.g. the PCIe End Point AMC module has to be activated before the Root Complex AMC module but the back plane FRU data does not support that.		
pwr	on|off <num#>	Sets the payload power strategy for the AMC slot with number <num#> = 1.... on / off means to be powered permanently / automatically.		R
		Displays the present state of the slot's payload power (powered / ---) and the selected payload power strategy (permanent / automatic)		R

Table 6:   CLI Command List

Command	Parameters	Description	User or Root
rs	[nvparamclear]	'rs' resets the MCH (cold reset). The current operating states (M states) of the AMC boards are not affected. 'rs nvparamclear' additionally clears all current configuration options. To adapt the MCH to its hardware and environment again you must close the Hot Swap switch or execute a manual configuration.	R
showerror		Displays the error variable (hex) and error text in detail	UR
showid		Display the IDs which were used to identify the MCH hardware and the backplane. For the MCH hardware this is the 'Board Product Name' in FRU 0 data (MCH's CM FRU data). For the backplane this is the 'Board Part Number' in FRU 253 (Backplane FRU data). If no FRU data is readable the value 'Default' will be displayed.	UR
showms		Show the current M states for all FRUs in the rack	UR
show-shaddr		Display the shelf address being read at MCH start. Please note that this value will not be refreshed until the next MCH start.	UR
slot	on|off <num#>	Sets the AMC slot with number <num#> = 1... to state AMC managed / AMC not managed i.e. to state AMC present / AMC not present. Note: Only applicable in racks where the presence of AMC modules is not detected automatically, e.g. in OM6062. In racks where PS1 signals are available the slot state will be set automatically.	R
		Displays the present slot state on / off (present / --) of all slots	R
sreg	rd <Page> <Reg> <NUM>	Eth. Switch Register access read command. <Page> : Page number <Reg> : Register offset <NUM>: Number of bytes to read.	R
	wr <Page> <Reg> <Data>*	Eth. Switch Register access write command. <Page> : Page number <Reg> : Register offset <Data> : 1 to 8 data byte(s), LS Byte first.	R

Table 6:   **CLI Command List**

Command	Parameters	Description	User or Root
tcp	\<initialFanLevel\> \<minFanLevel\> \<coolingFanIncrease-Timeout\> \<coolingFanDecrease-Timeout\> \<fanLevelStepUp\> \<fanLevelStepDown\> \<normalStableTime\>	Set temperature control parameters (hex)	R
		Display the current temperature control parameters (hex). Example: ```Read temperature control parameters (hex): initialFanLevel = 08 minFanLevel = 03 coolingFanIncreaseTimeout = 3C coolingFanDecreaseTimeout = 3C fanLevelStepUp = 02 fanLevelStepDown = 01 normalStableTime = 0708```	R
test	5	Test the watchdog. If the watchdog is configured after 10 seconds a reset must be executed.	R
tipmi	\<NetFn\> \<Lun\> \<Cmd\> [\<Byte\>*]	Send a raw IPMI command to the MCH's MMC. \<NetFn\> : Network function code \<Lun\> : Logical unit number \<Cmd\> : IPMI command \<Byte\> : Request data byte(s)	R
tipmb	\<SA\> \<NetFn\> \<Lun\> \<Cmd\> [\<Byte\>*]	Send a raw IPMI command (using Send Message) to a SlaveAddress on IPMB-L. \<SA\> : Slave Address \<NetFn\> : Network function code \<Lun\> : Logical unit number \<Cmd\> : IPMI command \<Byte\> : Request data byte(s)	R
upon		Enable firmware update. Warning: This will cancel normal operation immediately!	R

5.2 Ethernet Management Port

The Ethernet management interfaces gives access to the IPMI system and its IPMI commands and to the CLI using the RMCP/RMCP+ protocol. As a standard under Linux the free tool 'ipmitool' can be used to do the communication.

Please note: ICMP is not supported by the MCH.

The MCH supports:

- IPMI Over LAN (IOL) for invoking IPMI commands
- Serial Over LAN (SOL)
- Virtual LAN (VLAN)

From firmware version R31 on, a DIP switch on the solder side of the MCH board is used to activate or deactivate the MCH's Ethernet management port. On the AM4901 board it is switch SW3/1 and on the AM5901 board it is switch SW2/1. Set the switch to position "OFF" to activate or "ON" to deactivate the MCH's Ethernet management port.

Please refer to "Peculiarities and Restrictions for Releases, R31".

Before the Ethernet management port can be used it must be set up. This may be done using the CLI and a sequence of commands 'lan set 1…'. Please refer to the CLI command descriptions.

The CLI command:

```
lan print 1
```

will list all current basic LAN parameters. If no valid LAN parameters are set, the following commands may be used to set up the port:

```
lan set 1 ipaddr <nn.nn.nn.nn>

lan set 1 netmask <nn.nn.nn.nn>

lan set 1 macaddr <nn.nn.nn.nn.nn.nn>

lan set 1 access on
```

Please note that under a MCH firmware version < R29 the MAC address is not set automatically and must be set using the 'macaddr' command above.

There are more standard IPMI LAN parameters accessible when ipmitool is used, but the CLI's 'lan' command allows the setting all parameters for an initial use of the Ethernet management port. All LAN parameters are stored in a non-volatile way.

If VLAN support is needed the it can be set up by:

```
lan set 1 vlan <vlan ID>
```

where <vlan ID> must be > 0.

VLAN support can be disabled by:

```
lan set 1 vlan off
```

The LAN parameters will be saved and remembered by the MCH. This is important for the usage of IOL after switching on the MCH without having a connection to the CLI.

In the following the free Linux tool 'ipmitool' (version >= 1.8.10) is used to run IOL and SOL. For additional information to possible parameters and functionality please refer to the related online help, man page etc. of 'ipmitool'.

5.2.1 IPMI Over LAN (IOL) for Invoking IPMI Commands

IPMI Over LAN allows to invoke IPMI commands by using the Ethernet connection to the MCH's management port. More than one IOL connection may be active at one time.

For the usage of SOL only the LAN parameters must be set and activated. Please refer to the above.

A first user command after LAN set up might be similar to the following:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin mc info
```

This command returns the standard information about the running MCH firmware. The target is the Module Management Controller of the MCH.

If access to the Module Management Controller of an AMC module in the rack is required, the following parameters must be added to the command above before 'mc info':

```
ipmitool ... -t 0xnn -m 0x82 -b 7 mc info
```

where 'nn' is the address of the AMC module on the IPMB-L bus which is addressed by '-b 7'. For slots 1..6, '0xnn' is 0x72..0x7C.

If access to the Module Management Controller of a module on the IPMB-0 bus is required (e.g. a standard power module), the following parameters must be added, e.g. to the command above before 'mc info':

```
ipmitool ... -t 0xnn -m 0x82 -b 0 mc info
```

where 'nn' is the IPMB-0 address, e.g. of the standard power module. The address of the first standard power module is 0xC2. The address of the first standard cooling unit is 0xA8.

Hint: 'ipmitool' offers similar commands to display and set LAN parameters ('…lan set 1…', '…lan print 1').

5.2.2 Serial Over LAN (SOL) for CLI over LAN

For Serial Over LAN (SOL) the data exchange protocol being used by IOL is extended to transport any serial data instead of IPMI data (RMCP+ protocol).

If SOL is activated for the MCH, then received SOL data is transferred internally to the MCH's serial Command Line Interface (CLI). The response data of the CLI then is sent back via SOL. This allows running the CLI via Ethernet thus saving a physical serial connection between a terminal and the MCH's CLI.

For the activation of a SOL connection (i.e. using the CLI via Ethernet), the following SOL command must be invoked:

```
ipmitool -I lanplus -U admin -P admin -H 192.168.168.211
        -A NONE -C 0 -L USER sol activate
```

Please note that 'lanplus', i.e. the RMCP+ protocol, must be selected for that. Because the MCH does not support encryption, the '-C 0' parameter must be set.

The 'ipmitool' will report that SOL is activated:

```
[SOL session operational. Use ~? For help]
```

From now on until the SOL session is terminated, the keyboard may be used as if it were connected to the CLI. Hint: To obtain help, enter '~?'. Please note that the input '~' will not be reflected! Just continue with entering '?'.

Please note that standard CLI access via the serial management port is switched off during a SOL session and only one SOL session may be active at one time.

For the termination of an SOL connection enter the following:

```
~.
```

Please note that the input '~' will not be reflected. Just continue with entering '.'.

5.2.3 Virtual LAN (VLAN)

To set the VLAN ID (e.g. 3) for the MCH using the CLI enter the following:

```
lan set 1 vlan 3
```

To set the VLAN ID (e.g. 3) for the MCH via LAN enter the following:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin lan set 1
        vlan id 3
```

If the following is invoked quickly enough:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin lan print 1
```

the VLAN ID can be seen as being set to 3. After a few seconds further LAN access is no longer possible until the VLAN support on the host has been set up.

To set up a virtual LAN(e.g. 3) on a Linux host system requires root user and the module 8021q must be loaded. If it is not loaded, invoke the following:

```
modprobe 8021q
```

Assuming an Ethernet connection at "eth1", a virtual interface with the ID 3 can be created resulting in an "eth1.3" interface. This is done by invoking:

```
vconfig add eth1 3
```

Further, the IP address of "eth1" must be removed:

```
ifconfig eth1 0.0.0.0
```

and a valid ipaddress for eth1.3 specified and set up:

```
ifconfig eth1.3 192.168.168.212 up
```

A few seconds later, invoking the command:

```
arp -n
```

should indicate that the virtual LAN 3 is now available.

To use the virtual LAN 3, invoke for example:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin lan print 1
```

To switch off the VLAN support in the MCH, invoke for example:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin lan set 1
          vlan id off
```

or using the CLI, enter:

```
lan set 1 vlan off
```

To re-establish a virtual LAN, repeat the above as required.

6.   Configuration Parameters

6.1   Common Options (copts)

The first time the MCH is activated, the Common Options are automatically set to default values corresponding to the MCH board type. These options can be adapted by the user as necessary using the CLI command "copts" and the OEM IPMI commands "CmdGetCommonOptionsPaterameter" and the "CmdSetCommonOptionsParameter". All modifications remain in effect even after resetting or powering-down as long as when the MCH is re-activated no new board type is recognized or the non-volatile parameters of the MCH are lost.

Manual adaption of the options can be expedient for example in the following cases:

  • the MCH hardware is an AM4901 and the board revision is <R12

    Here the options bit 0 ("supprAM4901ErrorBeforeBoardRev12") should be set to suppress the issuing of an error message.

  • the automatic temperature regulation is to be deactivated via the parameter:

    tempCtrlDevice = 0

  • the modified options are to be permanently protected against automatic adaption.

    This protection is achieved by setting the options bit 31 ("locked").

Singular changes of options (bit-wise) are not possible. All Common Options must be read each time, modified where required, and then re-written.

The following table provides a listing of all Common Options.

Table 7:   Common Options List

Bits	Common Options	Format	Description	Set <u>A</u>utomatic / <u>M</u>anual
31	locked	bool	1 = All other Common Options are locked against automatic modifications. Only manual changes by e.g. CLI command 'copts' are possible. 0 = Not locked	M
30	suppressRedLEDForEvent-ProcessingError	bool	1 = suppress the red health LED in cases where only an event log processing error is active. This error occurs normally only if the event log is full. Note: The event log can be cleared by a standard IPMI command.	M
29:24	---	---	Reserved	---
23:21	tempCtrlDevice	---	ID of the used Temperature Control Strategy >1 = Undefined ID 1 = Temperature control is driven by temperature events. Automatic Fan Speed 0 = No automatic temperature control. Fan Speed can be set manually by CLI command 'fan'	A, M
20	processPwrShutdownEvent	bool	Some older IPMI firmware versions for AMC modules send misleading Power Shut Down events at start time. This behaviour can prevent the activation of the AMC module. 1 = Process all Power Shut Down Events at any time. 0 = Suppress the processing of Power Shut Down Events when not waiting for them e.g. when waiting for Quiesce state during deactivation. Please note that this suppression might prevent the deactivation of a module after a simple OS shut down because in this case a Power Shut Down will be simulated.	A, M
19	---	---	Reserved	---

Table 7:   Common Options List

Bits	Common Options	Format	Description	Set Automatic / Manual
18:16	phyDevice	---	ID of the used PHY device >2 = Undefined ID 2 = BCM5464 is selected (applicable e.g. on MCH AM5901) 1 = BCM5461 is selected (applicable e.g. on MCH AM4901) 0 = No PHY device is selected.	A, M
15:10	---	---	Reserved	---
9:7	prioMode	---	Is set explicit by CLI command "prioopts" and OEM IPMI command "Set Priority Options Parameters". Please refer to the descriptions of these commands. 0 = AUTO 1 = MANU 2 = OFF	A, M
6:4	activConfigMode	---	Is set explicit by CLI command "activopts" and OEM IPMI command "Set Activation Options Parameters". Please refer to the descriptions of these commands. 0 = AUTO 1 = MANU 2 = OFF	
3:1	pwrMngMode	---	Is set explicit by CLI command "currentopts" and OEM IPMI command "Set Current Options Parameters". Please refer to the descriptions of these commands. 0 = AUTO 1 = MANU 2 = OFF	
0	supprAM4901Error BeforeBoardRevR12	bool	On AM4901 boards with revision < R12 there is no MDIO interface assembled. Setting this bit to 1 prevents setting the red Health LED for errors being caused by this fact. Please note that the errors nevertheless are stored in the error variable as initialization error for the PHY device or runtime error for the switch device.	M

6.1.1 Defaults

Each MCH has its own set of default values for Common Options. In the event that no MCH hardware is recognized, the values for the AM4901 are used.

6.1.1.1 MCH type Not Recognized

In the event that no MCH hardware is recognized, the values for the MCH are as following:

$$
\begin{aligned}
locked &= 0 \\
suppressRedLEDForEventProcessingError &= 0 \\
tempCtrlDevice &= 1 \\
processPwrShutdownEvent &= 1 \\
phyDevice &= 0 \\
prioMode &= 0 \\
activConfigMode &= 0 \\
pwrMngMode &= 0 \\
supprAM4901ErrorBeforeBoardRevR12 &= 0
\end{aligned}
$$

6.1.1.2   AM4901

The following is a list of the default Common Options for the AM4901 MCH:

```
                          locked  =  0
suppressRedLEDForEventProcessingError  =  0
                   tempCtrlDevice  =  1
            processPwrShutdownEvent  =  1
                        phyDevice  =  1
                         prioMode  =  0
                   activConfigMode  =  0
                      pwrMngMode  =  0
    supprAM4901ErrorBeforeBoardRevR12  =  0
```

6.1.1.3   AM5901

The following is a list of the default Common Options for the AM5901 MCH:

```
                          locked  =  0
suppressRedLEDForEventProcessingError  =  0
                   tempCtrlDevice  =  1
            processPwrShutdownEvent  =  1
                        phyDevice  =  2
                         prioMode  =  0
                   activConfigMode  =  0
                      pwrMngMode  =  0
    supprAM4901ErrorBeforeBoardRevR12  =  0
```

6.2   Environmental Options (eopts)

At the first activation of the MCH (Hot Swap switch closed) the Environmental Options are set automatically to default values corresponding to the type of backplane in the system. The user can read these options, and if necessary they can be modified with the CLI command "eopts" or the OEM IPMI commands "CmdGetEnvironOptionsParameter" and "CmdSetEnvironOptionsParameter". All modifications remain in effect even after resetting or powering-down as long as when the MCH is re-activated no new board type is recognized or the non-volatile parameters of the MCH are lost.

Manual adaption of the options can be expedient for example in the following case:

- the modified options are to be permanently protected against automatic adaption.

    This protection is achieved by setting the options bit 31 ("locked").

Singular changes of options (bit-wise) is not possible. All Common Options must be read each time, modified where required, and then re-written.

The following table provides a listing of all Environmental Options.

Table 8:   Environmental Options List

Bits	Environmental Options	Format	Description	Set Automatic / Manual
31	locked	bool	1 = All Environmental Options in bits 30 .. 0 are locked against automatic modifications during adaption to a new rack. Only manual changes by e.g. CLI command 'eopts' are possible. 0 = Not locked against automatic modification	A, M
30:23	---	---	Reserved	---
22:20	fclockMode	---	Selected mode of fclock e-keying 0 = Fclock generated by the backplane Always on or switched on when PCIe is active. Only port and no clock e-keying for modules). 1 = Slot specific manual selection of fclock strategy and topology. 2 = AMC/uTCA specification compliant Fclock e-keying. Note: Automatically set when MCH detects the backplane type only if not 'locked'. Always automatically set by CLI command 'fclockopts' and by OEM IPMI command 'Set Fclock Options Parameter'.	A, M
19	pwrChanPlus4IfIs1ForAmc1	bool	1 = Corrects the backplane FRU data of OM6040 type racks	A, M

Table 8:   Environmental Options List

Bits	Environmental Options	Format	Description	Set Automatic / Manual
18:16	shAddrDevice	---	ID of the used mode for access to the shelf address/carrier number >3 = Undefined ID 3 = Shelf address/carrier number will be read as byte at I2C address 0x7C (e.g. on OM6060). The value will be inverted and 1 added (from R28 on). Results in value 1…16. Adjustable by back plane switch 2.1 … 2.4. 2 = Shelf address/carrier number will be read as byte at I2C address 0x4C (e.g. on ELMA K1 and K2). The value is the raw register contents. 1 = Shelf address/carrier number will be read as byte at I2C address 0x7C (e.g. on OM6062). The value will be inverted and 1 added (from R28 on). Results in value 1…16. Normally defined by the appropriate setting of a back plane switch. For further information, refer to the User Guide for the system in question. 0 = No shelf address/carrier number will be accessed	A, M
15	---	---	Reserved	---
14:12	ps1EnDevice	---	ID of the strategy to detect a seated AMC module >3 = Undefined ID 3 = Like 1 but with 4 slots 2 = Automatic detection by a Standard MTCA intelligent power device 1 = Automatic detection by a PCA8574A e.g. on OM6060 6 slots 0 = No automatic detection e.g. on OM6062. Managed modules i.e. slots have to be defined by e.g. CLI command 'slot'	A, M

Table 8:   Environmental Options List

Bits	Environmental Options	Format	Description	Set <u>A</u>utomatic / <u>M</u>anual
11:8	clkDevice	---	ID of the used clock device >2 = Undefined ID 2 = ICSFG108 switched on e.g. OM6060 1 = ICS9F3108 default state on e.g. OM6062 0 = No clock device defined	A, M
7:4	pwrDevice	---	ID of the used power device >4 = Undefined ID 4 = Like 2 but with 4 slots 3 = Standard MTCA intelligent power device 2 = TPS2359 on e.g. OM6060, 6 slots 1 = PCA9555 on e.g. OM6062 0 = No power device defined	A, M
3:0	fanDevice	---	ID of the used fan device >5 = Undefined ID 5 = Like 2 but with table 1 4 = MAX6651 with fan speed inverted and on/off mode on e.g. new 'short' OM6060 3 = Standard MTCA intelligent fan device 2 = MAX6651 with fan speed inverted on e.g. new 'short' OM6062R, table 0 1 = MAX6651 on e.g. old OM6062 0 = No fan device defined	A, M

6.2.1   Defaults

Each system has its own set of default values for Environmental Options. For the meaning of the settings please refer to table 8 "Environmental Options List" above.

6.2.1.1   Backplane Not Recognized or Standard Backplane/Rack

The following is a list of the default Environmental Options for an unknown backplane or a standard backplane/rack:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 2 \\
\text{pwrChanPlus4Ifls1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 2 \\
\text{clkDevice} &= 0 \\
\text{pwrDevice} &= 3 \\
\text{fanDevice} &= 3
\end{aligned}
$$

PCI Express communication will work if the required slot-to-slot connections are on the backplane. Standard Fclock e-keying is preselected (fclockopts 'spec'). If the Fclock is permanently supplied by the backplane then fclockopts 'rack' must be selected. Please refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

6.2.1.2   Backplane Type OM6062 Recognized

The following is a list of the default Environmental Options for the OM6062:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 0 \\
\text{pwrChanPlus4Ifls1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 0 \\
\text{clkDevice} &= 1 \\
\text{pwrDevice} &= 1 \\
\text{fanDevice} &= 1
\end{aligned}
$$

PCI Express is fully supported because the Fclock is supplied by the backplane. All AMC modules which require PCIe communications are expected to be an Fclock receiver requiring no Fclock e-keying. If Fclock e-keying is required, then select fclock mode 'manu' and its parameters and refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

6.2.1.3 Backplane Type OM6062 Short Recognized

The following is a list of the default Environmental Options for the OM6062 short:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 0 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 0 \\
\text{clkDevice} &= 1 \\
\text{pwrDevice} &= 1 \\
\text{fanDevice} &= 2
\end{aligned}
$$

PCI Express is fully supported because the Fclock is supplied by the backplane. All AMC modules which require PCIe communications are expected to be an Fclock receiver requiring no Fclock e-keying. If Fclock e-keying is required, then select fclock mode 'manu' and its parameters and refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

6.2.1.4 Backplane Type ELMA K1 or K2 Recognized

The following is a list of the default Environmental Options for the OM6062 short:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 0 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 2 \\
\text{ps1EnDevice} &= 0 \\
\text{clkDevice} &= 1 \\
\text{pwrDevice} &= 1 \\
\text{fanDevice} &= 1
\end{aligned}
$$

PCI Express is fully supported because the Fclock is supplied by the backplane. All AMC modules which require PCIe communications are expected to be an Fclock receiver requiring no Fclock e-keying. If Fclock e-keying is required, then select fclock mode 'manu' and its parameters and refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

6.2.1.5   Backplane Type OM6060 Recognized

The following is a list of the default Environmental Options for the OM6060:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 0 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 3 \\
\text{ps1EnDevice} &= 1 \\
\text{clkDevice} &= 2 \\
\text{pwrDevice} &= 2 \\
\text{fanDevice} &= 4
\end{aligned}
$$

PCI Express is fully supported because the Fclock is supplied by the backplane. All AMC modules which require PCIe communications are expected to be an Fclock receiver requiring no Fclock e-keying. If Fclock e-keying is required, then select fclock mode 'manu' and its parameters and refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

6.2.1.6   Backplane Type OM6040D index 0 and index 1 Recognized

The following is a list of the default Environmental Options for the OM6040D index 0/1:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 2 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 3 \\
\text{ps1EnDevice} &= 3 \\
\text{clkDevice} &= 0 \\
\text{pwrDevice} &= 4 \\
\text{fanDevice} &= 5
\end{aligned}
$$

PCI Express is not fully supported by this MCH because the Fclock is neither supplied by the backplane nor by the MCH. The Fclock mode is preset to 'spec' i.e. a spec conformant clock e-keying is preselected. PCI express will work if the Fclock is supplied by a processor AMC and both processor AMC and I/O AMC are able to process clock e-keying. If not refer to CLI command 'fclockopts' and chapter 3.5 E-keying. Select fclock mode 'manu' and its parameters.

6.2.1.7   Backplane Type OM6120 Recognized

The following is a list of the default Environmental Options for the OM6120:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 1 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 2 \\
\text{clkDevice} &= 0 \\
\text{pwrDevice} &= 3 \\
\text{fanDevice} &= 2
\end{aligned}
$$

PCI Express communication will not work with this MCH because the Fclock is neither supplied by the backplane nor by the MCH. There are no direct slot-to-slot PCIe connections on the backplane

6.2.1.8   Backplane Type OM6040

The following is a list of the default Environmental Options for the OM6040:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 1 \\
\text{pwrChanPlus4IfIs1ForAmc1} &= 1 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 2 \\
\text{clkDevice} &= 0 \\
\text{pwrDevice} &= 3 \\
\text{fanDevice} &= 0
\end{aligned}
$$

PCI Express communication will not work with this MCH because the Fclock is neither supplied by the backplane nor by the MCH. There are no direct slot-to-slot PCIe connections.

6.2.1.9   Backplane Type OM6063

The following is a list of the default Environmental Options for the OM6063:

$$
\begin{aligned}
\text{locked} &= 0 \\
\text{fclockMode} &= 0 \\
\text{pwrChanPlus4Ifls1ForAmc1} &= 0 \\
\text{shAddrDevice} &= 1 \\
\text{ps1EnDevice} &= 2 \\
\text{clkDevice} &= 0 \\
\text{pwrDevice} &= 3 \\
\text{fanDevice} &= 3
\end{aligned}
$$

PCI Express is fully supported because the Fclock is supplied by the backplane. All AMC modules which require PCIe communications are expected to be an Fclock receiver requiring no Fclock e-keying. If Fclock e-keying is required, then select fclock mode 'manu' and its parameters and refer to CLI command 'fclockopts' and chapter 3.5 E-keying.

7. Sensors

Sensors (software objects) record and administer operational states such as voltages, temperatures, and other parameters of the MCH and AMC modules. Sensors effect certain events when changes occur which result in the recording of entries in the System Eventlog.

For each sensor there is a Sensor Data Record (SDR) which describes the sensor type, the sensor name, limit values, hysteresis, unit, etc. Furthermore, the implemented possibilities for event origination are defined.

7.1 MCH Sensors

The MCH has sensors for measuring its own voltages, temperatures, and other states. Other MCH sensors record for example the Hot Swap state of all controlled AMC modules.

For OEM (Kontron) specific sensor types and reading types in the following table, refer to the next chapter.

Table 9: MCH Sensor List

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*] / Assert/deassert Mask
00h / FRU0 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor MCH	N
01h / AMC1 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 1. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N
02h / AMC2 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 2. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N
03h / AMC3 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 3. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N

Table 9: **MCH Sensor List**

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*)] / Assert/deassert Mask
04h / AMC4 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 4. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N
05h / AMC5 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 5. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N
06h / AMC6 Hot Swap	ATCA/CTCA Hot Swap (F0h) / Sensor-specific (6Fh)	00FFh / 0000h / 00FFh	Hot swap sensor AMC module in slot 6. Sensor is defined (visible e.g. using 'ipmitool') only when module is present	N
07h / FRU0 Reconfig	System Event (12h) Sensor-specific (6Fh)	0001h / 0001h / 0001h	System Event, e.g. when sensor list was modified during activation of a module.	N
08h / Temp Air Inlet	Temperature (01h) / Threshold (01h)	7A95h / 7A95h / 3F3Fh	Board Air Temperature	Y / 0F3Ch / 0000h
09h / Power Good	Power supply (08h) / OEM (73h)	0000h / 0000h / 001Fh	Status of all power lines	N
0Ah / Pwr Good Evt	Power supply (08h) / OEM (73h)	0000h / 001Fh / 001Fh	Power fail events for all power lines	Y / 0000h / 00FFh
0Bh / Vcc +12V	Voltage (02h) / Threshold (01h)	2204h / 2204h / 1212h	Board 12V supply	Y / 0F3Ch / 0000h
0Ch / Vcc +3.3V	Voltage (02h) / Threshold (01h)	2204h / 2204h / 1212h	Board 3.3V supply	Y / 0F3Ch / 0000h

Table 9:   **MCH Sensor List**

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*)] / Assert/deassert Mask
0Dh / Vcc +3.3V MMC	Voltage (02h) / Threshold (01h)	2204h / 2000h / 1212h	Board 3.3V supply (MP)	Y / 0F3Ch / 0000h
0Eh / Vcc +1.2V	Voltage (02h) / Threshold (01h)	2204h / 2204h / 1212h	Board 1.2V supply	Y / 0F3Ch / 0000h
0Fh / IPMI Watchdog	Watchdog2 (23h) / Sensor-specific (6Fh)	010Fh / 0000h / 010Fh	IPMI Watchdog	Y / 010Fh / 0000h
10h / IPMB0 Link State	IPMB-0 status change (F1h) / Sensor-specific (6Fh)	000Fh / 0000h / 000Fh	IPMB-0 state (refer to PICMG 3.0 Rev 2.0, 3.8.4.1)	N
11h / FRU0 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State	N
12h / AMC1 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 1 Module	Y
13h / AMC2 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 2 Module	Y
14h / AMC3 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 3 Module	Y
15h / AMC4 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 4 Module	Y
16h / AMC5 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 5 Module	Y
17h / AMC6 IPMBL State	IPMB-L status change (C3h) / Sensor-specific (6Fh)	000Ch / 0000h / 000Ch	IPMB-L State of AMC 6 Module	Y

Table 9:   MCH Sensor List

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*)] / Assert/deassert Mask
18h / Health Error	Platform Alert (24h) / 'digital' Discrete (03h)	0000h / 0000h / 0003h	Aggregates states (voltage threshold or temperature threshold exceeded, watchdog event). Visualization by the Health LED.	N
19h / AMC1 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 1 Module	N
1Ah / AMC2 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 2 Module	N
1Bh / AMC3 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 3 Module	N
1Ch / AMC4 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 4 Module	N
1Dh / AMC5 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 5 Module	N
1Eh / AMC6 Sensor Err	Management Subsystem Health (028h) / Sensor-specific (6Fh)	0001h / 0000h / 0001h	Management Subsystem Health of AMC 6 Module	N
1Fh / FRU0 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	FRU 0 Power Denied 1 = o.k., no alert, power not denied	N
20h / AMC1 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 1 Module Power Denied 1 = o.k., no alert, power not denied	N
21h / AMC2 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 2 Module Power Denied 1 = o.k., no alert, power not denied	N

Table 9:   MCH Sensor List

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*)] / Assert/deassert Mask
22h / AMC3 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 3 Module Power Denied 1 = o.k., no alert, power not denied	N
23h / AMC4 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 4 Module Power Denied 1 = o.k., no alert, power not denied	N
24h / AMC5 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 5 Module Power Denied 1 = o.k., no alert, power not denied	N
25h / AMC6 Pwr Denied	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0002h / 0003h	AMC 6 Module Power Denied 1 = o.k., no alert, power not denied	N
26h / FRU0 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	FRU 0 State	N
27h / AMC1 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 1 Module State	N
28h / AMC2 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 2 Module State	N
29h / AMC3 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 3 Module State	N
2Ah / AMC4 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 4 Module State	N

Table 9:   MCH Sensor List

SENSOR Number / ID string	SENSOR TYPE (CODE) / EVENT/ READING TYPE (CODE)	Ass. Mask / Deass. Mask / Reading Mask	DESCRIPTION	Causes red Health LED on error [*] / Assert/deassert Mask
2Bh / AMC5 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 5 Module State	N
2Ch / AMC6 Agent	FRU Agent (C5h) Discrete (0Ah)	0140h / 0000h / 0147h	AMC 6 Module State	N
2Dh / IPMC Storage Err	Management Sub-system Health (28h) / Sensor-specific (6Fh)	0002h / 0000h / 0003h	IPMI controller storage access error	N
2Eh / IPMC Reboot	Platform Alert (24h) / 'digital' Discrete (03h)	0002h / 0000h / 0003h	2 = Management controller is (re-)booting	N
2Fh / Ver change	Firmware version change (2Bh) / Sensor-specific (6Fh)	0002h / 0000h / 0002h	Management controller's firmware version changed	N
30h / SEL State	Event Logging Disabled (10h) / Sensor-specific (6Fh)	003Ch / 0000h / 003Ch	State of event logging	N
31h / IPMI Info-1	OEM Firmware Info 1 (C0h) / OEM (70h)	0003h / 0000h / 7FFFh	For internal use only	N
32h / IPMI Info-2	OEM Firmware Info 2 (C0h) / OEM (70h)	0003h / 0000h / 7FFFh	For internal use only	N

7.2   OEM (Kontron) Sensors

The following table provides a list of OEM (Kontron) sensors.

Table 10:   OEM (Kontron) Sensors List

OEM SENSOR TYPE (CODE)	OEM EVENT/ READING TYPE (CODE)	DESCRIPTION	
Firmware Info 1 (C0h)	70h	Internal Diagnostic Data	
Firmware Info 2 (C0h)	71h	Internal Diagnostic Data	
Initialization Agent (C2h)	03h ('digital' Discrete)	Offsets / events: 0: Initialization O.K. 1: Initialization Error	
IPMBL State (C3h)	6Fh (sensor type specific)	Sensor discrete State	Meaning
		08h	IPMB-L running
		others	IPMB-L not running
FRU Agent (C5h)	0Ah (Discrete)	FRU initialization agent, using a standard reading type.	
Firmware Upgrade Manager (C7h)	6Fh (sensor type specific)	Offsets / events: 0 : First Boot after upgrade 1 : First Boot after rollback (error) 2 : First Boot after errors (watchdog) 3 : First Boot after manual rollback 4 : Reserved 5 : Reserved 6 : Reserved 7 : Reserved 8 : Firmware Watchdog Bite, reset occurred	

Table 10:   OEM (Kontron) Sensors List

OEM SENSOR TYPE (CODE)	OEM EVENT/ READING TYPE (CODE)	DESCRIPTION	
Board Reset (CFh)	03h ('digital' Discrete)	Data 2 contains the reset type: ...WARM = 0 ...COLD = 1 ...FORCED_COLD = 2 ...SOFT_RESET = 3 ...MAX = 4 Data 3 contains the reset source: ...IPMI_WATCHDOG = 0 ...IPMI_COMMAND = 1 ...PROC_INT_CHECKSTOP = 2 ...PROC_INT_RST = 3 ...RESET_BUTTON = 4 ...POWER_UP = 5 ...LEG_INITIAL_WATCHDOG = 6 ...LEG_PROG_WATCHDOG = 7 ...SOFTWARE_INITIATED = 8 ...SETUP_RESET = 9 ...UNKNOWN = 0xFF	
e.g. for Power Good / Power Good Event	73h	Sensor-specific Offset	Event Bit set = o.k.
		0h	Vcc Core good
		1h	Vcc 1.2V good
		2h	Vcc 3.3V MP good
		3h	Vcc 3.3V good
		4h	Vcc 12V good
Hot swap sensor (F2h)	6Fh (sensor type specific)	Sensor-specific Offset	Event
		00h	Handle closed
		01h	Handle open
		02h	Quiesced

7.3 MCH Analog Sensor Thresholds

The following table indicates the analog sensor thresholds for the MCH.

Table 11:   MCH Analog Sensor Thresholds

SENSOR Number / ID string	Lower non recoverable	Lower critical	Lower non critical	Normal min	Nominal	Normal max	Upper non critical	Upper critical	Upper non recoverable
08h / Temp Air Inlet AM4901 / / AM5901	-45 °C	-40 °C	-35 °C	5 °C	25 °C	35 °C	55 °C	60 °C	70 °C
0Bh / Vcc +12V	n.a.	10.64V	n.a.	11.4 V	12.0 V	12.6 V	n.a.	13.27 V	n.a.
0Ch / Vcc +3.3V	n.a.	3.12 V	n.a.	3.12 V	3.3 V	3.46 V	n.a.	3.46 V	n.a.
0Dh / Vcc +3.3V MMC	n.a.	3.12 V	n.a.	3.12 V	3.3 V	3.46 V	n.a.	3.46 V	n.a.
0Eh/ Vcc +1.2V	n.a.	4.66 V	n.a.	4.70 V	5.0 V	5.31 V	n.a.	5.36 V	n.a.

7.4   Sensor Data Record Repository

The currently valid sensors for the whole system are administered in the so called Sensor Data Record Repository (SDRR). The MCH's sensors are always valid as well as those of the AMC modules as long as they are active. When AMC modules are activated or deactivated, the SDRR is in a continuous state of flux. The order in which the AMC sensors are registered and the sensor numbers are subject to change.

Access to the SDRR via the system management software can be effected best from an AMC processor module over the IPMB-L bus using the free software tool "ipmitool". In the SDRR the sensor names for Kontron AMC modules have a prefix to identify the corresponding AMC slot in which they are installed. The prefixes A1, A2, A3, A4, B1 and B2 designate the AMC modules in the AMC slots 1 to 6 corresponding to FRU IDs 5 to 10. The names of the MCH sensors do not have prefixes. MCH sensors are always present, and therefore the first sensors inside the SDRR listing. As such it is easy to associate them with the MCH.

8.   Supported OEM IPMI Commands and Command Extensions

8.1   Extended Standard Commands

8.1.1   Get Device ID Command with Kontron OEM Extensions

COMMAND	LUN	NetFn	CMD
Get Device ID command with Kontron OEM extensions	00h	App = 06h	01h
REQUEST DATA			
Byte	Data Field		
-	-		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2:12	Regular Get Device ID Command response fields		
13	Release number 1… of the IPMI controller firmware, e.g. the open ipmi tool 'ipmitool' displays this as 'SDR' in the answer on command 'ipmitool fwum status'.		
14	Module Geographical Address (slot number): 1 … = Module in chassis slot 1… Please note that the MCH always reports 0.		
15	Developer's maintenance revision: 0x0 or 0xff identifies a released standard version 0x80…0xfe identifies a released customer version 0x01…0x7f identifies a version under development		
16	Reserved		

8.1.2   Master Write-Read Command

The IPMI Master Write-Read command normally supports access to IPMB busses/channels only in board test mode. To switch on the board test mode please refer to IPMI command 'Manufacturing Test On Command'.

In the table below, only the Kontron extension/modification of the standard command is provided. Refer to the IPMI specification for the complete description of the command.

COMMAND		LUN	NetFn	CMD
Master Read-Write Command		00h	App = 06h	52h
REQUEST DATA				
Byte	Data Field			

1	Bus ID is composed as follows:
	  [7:4] = 0 = for bus with carrier's EEPROM
	       = 1 = for bus with FRU/NVParams etc. EEPROM etc.
	       = 2 = for bus with Temperature sensors etc.
	       = 3 = for Inter MCH IPMB bus
	       = 4 = for IPMB-L
	       = 5 = for IPMB-A
	       = 6 = for IPMB-B
	  [3:1] = 1 = bus ID
	    [0] = 1 = bus type			

RESPONSE DATA				
Byte	Data Field			
1	Completion code			

8.1.3 Manufacturing Test On Command

In the table below, only the Kontron extension/modification of the standard command is provided. Refer to the IPMI specification for the complete description of the command.

COMMAND	LUN	NetFn	CMD
Manufacturing Test On Command	00h	App = 06h	05h
REQUEST DATA			
Byte	Data Field		
1:4	Password = 0xb4 0x90 0x91 0x8b		
5	Mode = 0x80 = Board test. The bits in byte 6 are valid only for the MCH		
6	Test options: switch on or off test disturbing activities [0] = 1 = Standard access to IPMBs is off [1] = 1 = Standard access to the I2C temperature device is off [2] = 1 = Standard sensor readings are off [3] = 1 = No standard MCH device is active (temp. control, power control etc.) [0:3] = 0 = All kinds of hardware access by the standard firmware is allowed, i.e. the test mode is completely disabled. Use this to disable the board test again.		
RESPONSE DATA			
Byte	Data Field		
1	Completion code		

8.2   Kontron OEM Commands

8.2.1   Write Switch Register

This command writes the register of a switch via the MIIM interface. The PHY Interface Address is not a parameter here because it is always the same default address.

COMMAND		LUN	NetFn	CMD
WriteSwitchRegister		00h	OEM = 3Eh	22h
REQUEST DATA				
Byte	Data Field			
1	Format = 0: for BCM5396 　　　　<>: undefined, error			
2	Register Page			
3	Register Offset			
4	Data Pattern LSB			
[5:11]	Higher Data Pattern Bytes (optional)			
RESPONSE DATA				
Byte	Data Field			
1	Completion Code			

8.2.2   Read Switch Register

This command reads the register of a switch via the MIIM interface. The PHY Interface Address is not a parameter here because it is always the same default address.

COMMAND		LUN	NetFn	CMD
ReadSwitchRegister		00h	OEM = 3Eh	23h
REQUEST DATA				
Byte	Data Field			
1	Format = 0: for BCM5396 　　　　<>: undefined, error			
2	Register Page			
3	Register Offset			
4	Byte Count n (1..8)			
RESPONSE DATA				
Byte	Data Field			
1	Completion Code			
2:n	Register data bytes. LSB first			

8.2.3   Set Fan Speed Parameter

This command writes the fan speed parameter to the related NV-Parameter, saves this to EE-PROM, and forces the new fan speed. It works only if automatic temperature control is inactive.

COMMAND	LUN	NetFn	CMD
SetFansSpeedParameter	00h	OEM = 3Eh	24h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	Fan speed level 0…0xf		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.4   Get Fan Speed Parameter

This command reads the current fan speed parameter.

COMMAND	LUN	NetFn	CMD
GetFansSpeedParameter	00h	OEM = 3Eh	25h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	Fan speed level 0…0xf		

8.2.5 Set Permanent Power Parameter

This command writes the permanent power parameter to the related NV-Parameter and saves it.

COMMAND	LUN	NetFn	CMD
SetPermPwrParameter	00h	OEM = 3Eh	26h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4:7	Permanent power parameter. LSB first. Note: AMC Slot 1 = bit 5		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.6 Get Permanent Power Parameter

This command gets the permanent power parameter from the related NV-Parameter.

COMMAND	LUN	NetFn	CMD
GetPermPwrParameter	00h	OEM = 3Eh	27h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5:8	Permanent power parameter. LSB first. Note: AMC Slot 1 = bit 5		

8.2.7 Set Slot Presence Parameter

This command writes the slot presence parameter to the related NV-Parameter and saves it.

COMMAND	LUN	NetFn	CMD
SetSlotPrsParameter	00h	OEM = 3Eh	28h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4:7	Slot presence parameter. LSB first. Note: AMC Slot 1 = bit 5		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.8 Get Slot Presence Parameter

This command gets the slot presence parameter from the related NV-Parameter.

COMMAND	LUN	NetFn	CMD
GetSlotPrsParameter	00h	OEM = 3Eh	29h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5:8	Slot presence parameter. LSB first. Note: AMC Slot 1 = bit 5		

8.2.9   Set Common Options Parameter

This command writes the common options parameter to the related NV-Parameter and saves it.

COMMAND	LUN	NetFn	CMD
SetCommonOptionsParameter	00h	OEM = 3Eh	2Ah
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4:7	Common options parameter. LSB first. Note: AMC Slot 1 = bit 5		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.10   Get Common Options Parameter

This command gets the common options parameter from the related NV-Parameter.

COMMAND	LUN	NetFn	CMD
GetCommonOptionsParameter	00h	OEM = 3Eh	2Bh
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5:8	Common options parameter. LSB first. Note: AMC Slot 1 = bit 5		

8.2.11 Set Slot Options Parameter

This command writes the common options parameter to the related NV-Parameter and saves it.

COMMAND		LUN	NetFn	CMD
SetSlotOptionsParameter		00h	OEM = 3Eh	2Ch
REQUEST DATA				
Byte	Data Field			
1	Kontron IANA LSB			
2	Kontron IANA byte 2			
3	Kontron IANA MSB			
4:15	Slot options parameter for slots 1…12. Slot 1 first.			
RESPONSE DATA				
Byte	Data Field			
1	Completion Code			
2	Kontron IANA LSB			
3	Kontron IANA byte 2			
4	Kontron IANA MSB			

8.2.12 Get Slot Options Parameter

This command gets the common options parameter from the related NV-Parameter.

COMMAND		LUN	NetFn	CMD
GetSlotOptionsParameter		00h	OEM = 3Eh	2Dh
REQUEST DATA				
Byte	Data Field			
1	Kontron IANA LSB			
2	Kontron IANA byte 2			
3	Kontron IANA MSB			
RESPONSE DATA				
Byte	Data Field			
1	Completion Code			
2	Kontron IANA LSB			
3	Kontron IANA byte 2			
4	Kontron IANA MSB			
5:16	Slot options parameter for slots 1…12. Slot 1 first.			

8.2.13 Clear All NV Parameters

This command clears/resets all NVPARAMs and saves them to EEPROM. Executes a cold reset if requested.

COMMAND	LUN	NetFn	CMD
ClearAllNvparam	00h	OEM = 3Eh	2Eh
REQUEST DATA			
Byte	Data Field		
1	0x99		
2	0x99		
3	0x99		
4	Reset request if != 0		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	0x99		
3	0x99		
4	0x99		

8.2.14 Set Password

This command writes the password to the related NV-Parameter and saves this to EEPROM.

COMMAND	LUN	NetFn	CMD
SetPassword	00h	OEM = 3Eh	2Fh
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4:12	Password, 0x0 terminated		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.15 Set Event Driven Temperature Control Parameter

This command sets the parameters for the event driven temperature control and saves them to EEPROM.

COMMAND	LUN	NetFn	CMD
SetEventDrivenTemperatureControlParameter	00h	OEM = 3Eh	30h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	initialFanLevel		
5	minFanLevel		
6	coolingFanIncreaseTimeout		
7	coolingFanDecreaseTimeout		
8	fanLevelStepUp		
9	fanLevelStepDown		
10	normalStableTime LSB		
11	normalStableTime MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.16   Get Event Driven Temperature Control Parameter

This command gets the parameters for the event driven temperature control.

COMMAND	LUN	NetFn	CMD
GetEventDrivenTemperatureControlParameter	00h	OEM = 3Eh	31h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	initialFanLevel		
6	minFanLevel		
7	coolingFanIncreaseTimeout		
8	coolingFanDecreaseTimeout		
9	fanLevelStepUp		
10	fanLevelStepDown		
11	normalStableTime LSB		
12	normalStableTime MSB		

8.2.17 Set Environmental Options Parameter

This command writes the environmental options parameter to the related NV-Parameter and saves it.

COMMAND	LUN	NetFn	CMD
SetEnvironOptionsParameter	00h	OEM = 3Eh	32h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4:7	Environmental options parameter. LSB first.		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.18 Get Environmental Options Parameter

This command gets the environmental options parameter from the related NV-Parameter.

Command	LUN	NetFn	CMD
GetEnvironOptionsParameter	00h	OEM = 3Eh	33h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5:8	Environmental options parameter. LSB first.		

8.2.19　Set Priority Options Parameters

This command sets the priority parameters and the related common options parameters and saves them to EEPROM.

Command	LUN	NetFn	CMD
CmdSetSlotPriosParameter (internal name)	00h	OEM = 3Eh	34h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	prioMode (refer to common options) 0=AUTO: Use parameters from back plane FRU data "MicroTCA Carrier Activation and Power Management record" 1=MANU: Use Slot priority parameters (refer to below) 2=OFF: No priorities used. Activation and deactivation sequence is random.		
5:8	reserved=0		
9:9+12	Slot priority parameter values of 12 slots. Beginning with slot 1. Use every priority value 1 … e.g. 6 only once. Low number = high priority. Priorities for not existing slots e.g. 7…12 are void but shall be unique as well. Bit 7 set additionally = uncoordinated slot, not member of sequence.		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.20   Get Priority Options Parameters

This command gets the currently active slot priority parameters and the related common options parameters.

Command	LUN	NetFn	CMD
CmdGetSlotPriosParameter (internal name)	00h	OEM = 3Eh	35h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	prioMode (refer to common options) Please refer to command "Set Priority Parameters" above.		
6:9	reserved=0		
10:10 +12	Slot priority parameter values of 12 slots Please refer to command "Set Priority Parameters" above.		

8.2.21   Set Activation Options Parameters

This command sets the activation and deactivation parameters and the related common options parameters and saves them to EEPROM.

Command	LUN	NetFn	CMD
CmdSetSlotActivConfigParameter (internal name)	00h	OEM = 3Eh	36h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	activConfigMode (refer to common options) 0=AUTO: Use parameters from back plane FRU data "MicroTCA Carrier Activation and Power Management record" 1=MANU: Use Slot activation/deactivation parameters (refer to below) 2=OFF: No delays during activation/deactivation used.		
5	allowReady [s] When a module needs more than this time for getting ready (M3) then the activation sequence is continued.		
6:8	reserved=0		
9:9+12	Delays before processing next slot [100ms] for 12 slots. Beginning with slot 1. Delays for not existing slots e.g. 7…12 are void. Maximal value is 3fh i.e. 6.3s.		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.22 Get Activation Options Parameters

This command gets the currently active activation and deactivation parameters and the related common options parameters.

Command	LUN	NetFn	CMD
CmdGetSlotActivConfigParameter (internal name)	00h	OEM = 3Eh	37h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	activConfigMode (refer to common options) Please refer to command "Set Activation Options Parameters" above.		
6	allowReady [s] Please refer to command "Set Activation Options Parameters" above.		
7:9	reserved=0		
10:10 +12	Delays before next slot [100ms] for 12 slots. Please refer to command "Set Activation Options Parameters" above.		

8.2.23   Set Current Options Parameters

This command sets the current parameters (limits) and the related common options parameters and saves them to EEPROM.

Command	LUN	NetFn	CMD
CmdSetPwrMngConfigParameter (internal name)	00h	OEM = 3Eh	38h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	pwrMngMode (refer to common options) 0=AUTO: Use parameters from back plane FRU data "MicroTCA Carrier Power Policy record" 1=MANU: Use current options parameters (refer to below) 2=OFF: No current limits used.		
5:6	maxCurrentOverride [100mA], LSB first. Maximal current which the rack's power supply is able to deliver.		
7:8	reserved=0		
9:9+12	Maximal allowed current consumption [100mA] for 12 slots. Beginning with slot 1. Values for not existing slots e.g. 7…12 are void.		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		

8.2.24 Get Current Options Parameters

This command gets the currently active slot activation and deactivation parameters and the related common options parameters.

Command	LUN	NetFn	CMD
CmdGetPwrMngConfigParameter (internal name)	00h	OEM = 3Eh	39h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	pwrMngMode (refer to common options) Please refer to command "Set Current Options Parameters" above.		
6:7	maxCurrentOverride [100mA] Please refer to command "Set Current Options Parameters" above.		
8:9	reserved=0		
10:10 +12	Maximal allowed current consumption [100mA] for 12 slots. Please refer to command "Slot Set Current Options Parameters" above.		

8.2.25   Set Fclock Options Parameter

This command sets the Fclock options parameters of six slots to the related NV-Parameters and environmental options and saves them to EEPROM. Parameter slotGroup = 0 / > 0 selects the first / second group of six slots. Also the related fclockMode bits in the environmental options will be set and saved to EEPROM.

Command	LUN	NetFn	CMD
CmdSetFclockOptionsParameter (internal name)	00h	OEM = 3Eh	3Ah
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	slotGroup 0 = Slots 1...6 >0 = Slots 7...12		
5	fclockMode FCLOCK_MODE_RACK = 0 = Fclock generated by the backplane. Always on or switched on when PCIe is active. Only port and no clock e-keying for modules. FCLOCK_MODE_MANU = 1 = Slot specific manual selection of fclock strategy and topology FCLOCK_MODE_SPEC = 2 = AMC/uTCA specification compliant Fclock e-keying		
6:7	reserved = 0		

Command	LUN	NetFn	CMD
CmdSetFclockOptionsParameter (internal name)	00h	OEM = 3Eh	3Ah
REQUEST DATA (CONTINUED)			

Byte	Data Field
8:19	Board manual fclock options (remoteSlotNumber, fClockStrategy) of 6 slots.
	These parameters are only valid for FCLOCK_MODE_MANU.
	Usage of parameter 'Remote slot number':
	Only if for at least one source slot a 'remote slot number' > 0 is defined, the backplane's clock ptp FRU data will be replaced for ALL slots by FRU data being defined by all pairs of 'source slot number' and 'remote slot number' parameters.
	Remote slot number 0 means no remote slot connected. Each use of a remote slot number (except 0) must be unique, i.e. every slot may be connected to one remote slot only.
	Usage of parameter 'fclock strategy':
	'Fclock strategy' if 'remote slot number' is defined (1..) =
	1 = AMC Fclock parameters as specified in FRU data. Use matching ones.
	2 = AMC Fclock shall be receiver. Copy/modify remote AMC's parameters.
	3 = AMC Fclock shall be source. Copy/modify remote AMC's parameters.
	'Fclock strategy' if no 'remote slot number' is defined (0), AMC shall be source/receiver/ fixed e.g. by switch or default state =
	0 = Do not change AMC's Fclock. Use AMC's fixed or default state."
	1x = AMC Fclock shall be receiver as defined in xth match 0 descriptor.
	2x = AMC Fclock shall be source as defined in xth match 1 descriptor.
RESPONSE DATA	

Byte	Data Field
1	Completion Code
2	Kontron IANA LSB
3	Kontron IANA byte 2
4	Kontron IANA MSB

8.2.26 Get Fclock Options Parameter

This command gets the Fclock options parameters for six slots from the related NV-Parameters and environmental options. An offset parameter of 0/>0 selects the first or second group of slots (1...6 or 7...12).

Command	LUN	NetFn	CMD
CmdGetFclockOptionsParameter (internal name)	00h	OEM = 3Eh	38h
REQUEST DATA			
Byte	Data Field		
1	Kontron IANA LSB		
2	Kontron IANA byte 2		
3	Kontron IANA MSB		
4	slotGroup (refer to command Set Fclock Options Parameter)		
RESPONSE DATA			
Byte	Data Field		
1	Completion Code		
2	Kontron IANA LSB		
3	Kontron IANA byte 2		
4	Kontron IANA MSB		
5	slotGroup (refer to command Set Fclock Options Parameter)		
6	fclockMode (refer to command Set Fclock Options Parameter)		
7:8	reserved=0		
9:20	Board manual fclock options (remoteSlotNumber, fClockStrategy) of 6 slots (refer to command Set Fclock Options Parameter).		

9.   Basic Installation and Upgrade of the MCH Firmware

The firmware from version R29 on comes in three variants. They all are currently named "…AM4901…", but they are still common for both the AM4901 and the AM5901 MCH modules:

1. FW_IPMI_AM4901_FULL_Rxx.bin. This is the full firmware including boot block and application of version Rxx. It must be initially programmed during a basic installation using, for example, the tool 'lpcFlash' at the serial management interface.

2. FW_IPMI_AM4901_FWUM_Rxx.bin. This is the application firmware for upgrade of an already programmed firmware in binary format. The tool 'ipmitool' and its command 'ipmitool fwum …' is suitable for upgrading this firmware.

3. FW_IPMI_AM4901_FWUM_Rxx.hpm. This is the application firmware for upgrade of an already programmed firmware (1) in HPM.1 format. The tool 'ipmitool' and its command 'ipmitool hpm …' is suitable for upgrading this firmware.

9.1   Basic Installation

A basic installation of an MCH Firmware is possible for both MCH types AM4901 and AM5901. It is accomplished under Linux® via the utility program "lpcflash" which requires Root user rights. This requires a serial connection between the source computer and the MCH front panel serial interface. A basic installation is required:

- For the first firmware installation during manufacturing
- If important changes require it (e.g. upgrade from R29 to R30 or later)
- Always for an MCH of type AM4901.

Initiating the download process for basic installation can be done by issuing, for example, the following:

```
sudo lpcFlash -m am4901 -s 0 -b 115200 -e 9600
    FW_IPMI_AM4901_FULL_Rxx.bin
```

In the above example, the serial interface "/dev/ttyS0 (-s 0)" of the source computer is used for the data transfer. This interface may not be used concurrently with the terminal program that is being used to interface with the CLI. The terminal program must be terminated before starting the Firmware download. The download transfer baud rate is 115200 baud, and the CLI baud rate should be set to 9600 baud once the Firmware download is completed. In the example the Firmware to be downloaded is located in the file "FW_IPMI_AM4901_FULL_Rxx.bin".

Prior to starting "lpcFlash", switch 2 of SW3 on the AM4901 or SW2 on the AM5901 respectively must be set to OFF. Once the download is completed, this switch must be set to ON again.

If the MCH is to be installed in a rack of type OM6060 or OM6040D and the Ethernet management port is to be used, please disable the Management Ethernet port. Refer to chapter "11. Peculiarities and Restrictions for Releases" for R31.

If Firmware has been previously installed, then it is not necessary to change the switch settings prior to downloading. In this case instead, the CLI command "upon" can be issued to the MCH and the terminal program terminated prior to starting "lpcFlash". Once the download is completed, the terminal program can be restarted.

9.2 Upgrading

An upgrade of the firmware is only possible for an MCH of type AM5901 from R29 on. It is possible during full operation in the field. Please note that R29 -> R30 as an exception always requires first a basic installation (refer to above), then an upgrade to fill both firmware banks. The communication for the upgrade may be done using 'ipmitool'.

- Via IPMB-L using a processor AMC with KCS interface in the rack
- Via Ethernet using the MCH's management port

9.2.1 Upgrade via IPMB-L

On any processor AMC in the rack, an upgrade of the MCH is possible by using 'ipmitool' under Linux.

For upgrading with a .bin file, the command sequence is the following (nn is the IPMB-L address of the processor AMC):

```
ipmitool -t 0x20 -m 0xnn fwum download
        FW_IPMI_AM4901_FWUM_Rxx.bin

ipmitool -t 0x20 -m 0xnn fwum upgrade
```

For upgrading with a .hpm file the command sequence is the following (nn is the IPMB-L address of the processor AMC):

```
ipmitool -t 0x20 -m 0xnn hpm upgrade
        FW_IPMI_AM4901_FWUM_Rxx.hpm all

ipmitool -t 0x20 -m 0xnn hpm activate
```

9.2.2 Upgrade via Ethernet

From a remote system running Linux 'ipmitool' can upgrade the MCH firmware via an Ethernet connection to the management port. For that the management port has to be set up first. Please refer to the CLI command 'lan'.

For upgrading with a .bin file, the command sequence is similar to the following:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin
        fwum download FW_IPMI_AM4901_FWUM_Rxx.bin

ipmitool -I lan -H 192.168.168.211 -U admin -P admin fwum upgrade
```

For upgrading with a .hpm file, the command sequence is similar to the following:

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin hpm upgrade
        FW_IPMI_AM4901_FWUM_Rxx.hpm all
```

Please note: The parameter "all" must be used. This is required to preclude rejection of an upgrade, because ipmitool does not detect a version change.

```
ipmitool -I lan -H 192.168.168.211 -U admin -P admin hpm activate
```

10.   Special Failure Situations

10.1   Special Error Situations

The following information provides assistance to users for special error situations:

- Health LED is red: refer to chapter 3.8.3.

- AMC module does not activate: cause: slots improperly configured: refer to chapter 3.1.1, or the AMC's FRU data does not contain a Current Requirement Record

- MCH FRU data not readable: cause: Common Options corrupted: refer to chapter 6.1.1, re-enter and protect

- Backplane FRU data not readable: Environmental Options corrupted: refer to chapter 6.2.1, re-enter and protect

- Timeout during module activation occurs (message "WARNING: ACTIVATION TIMEOUT FOR MODULE IN SLOT… CHECK MODULE AND PARAMETERS!"). Please refer to CLI command 'activopts', chapter 5.1.2 CLI Command List.

- An AMC module can't be activated because it does not contain a Current Requirement Record in its FRU data or the current value is 0. Message "CURRENT REQUIREMENT RECORD MISSING FOR MODULE IN SLOT… OR VALUE IS 0." Or message "NO CARRIER POWER POLICY RECORD FOUND".

- An AMC module can't be activated because there is no related backplane FRU data. Message "CARRIER ACTIVATION AND CURRENT DESCRIPTOR MISSING FOR SLOT…".

- The MCH resets continuously after powering up. Probably the MCH runs in a rack of type OM6060 or OM6040D. Please disable the Management Ethernet port. Refer to chapter "11. Peculiarities and Restrictions for Releases" for R31.

10.2   Error Messages when Storage Overflow Occurs

When the backplane offers more data than the MCH is able to store, then one of the following error messages may occur. This should not occur in "well known" racks. Please note that it might be impossible to maintain all inserted AMC modules. The "0xnn" values indicated are the currently implemented limits.

```
ERROR: CARRIER_PTP_CONNECTIVITY_RECORD: Max PtP Res Descr
        exceeded >= 0x22
WARNING: CARRIER_PTP_CONNECTIVITY_RECORD: Max Port Descr
          exceeded = 0x21
WARNING: CARRIER_ADDRESS_TABLE_RECORD: Max Carrier Table
          exceeded = 0x10
WARNING: AMC_PTP_CONNECTIVITY_RECORD: Max AMC Channel Descr
          exceeded = 0x16
WARNING: AMC_PTP_CONNECTIVITY_RECORD: Max AMC Link Descr
          exceeded = 0x22
```

11.   Peculiarities and Restrictions for Releases

11.1   R27

1. The MCH can only be used in an OM6062 and ELMA K1 and K2 compatible chassis. I.e. it can't control MTCA specification conformant intelligent and redundant power units and cooling units.

2. No F-clock E-keying, according to AMC.0 R2. F-clocks for an AMC slot are automatically supplied when the slot's payload power is switched on, by backplane hardware.

3. The payload power is switched by Mosfet switches on a proprietary I2C bus. I.e. no intelligent power supply is supported.

4. No automatic presence detection for AMC modules. The population of the rack has once to be defined explicit by an operator via CLI during startup. I.e. no intelligent power supply is available

5. MCH firmware upgrade only by serial interface using a "Flash Tool" (lpcFlash, Flash Magic). No firmware backup and roll back to the former firmware version after a failed upgrade.

6. The power consumption of the AMC modules will not be taken to account in the power up phase of a module. Power will always be granted.

7. No clock E-keying is implemented. Please note that Kontron AMC modules until now do not need clock e-keying (AM400x, AM401x, AM410x, AM5010). Please note that OM6062 supplies every inserted module automatically with the F-clock.

8. The following FRU data records are not yet processed:

   • "MicroTCA Carrier Activation and Power Management Record"
   • "Module Current Requirement record" of the MCH and the AMC Modules
   • "MicroTCA Power Policy Record"
   • "MicroTCA Power Module Capability Record"
   • "MicroTCA FAN Geography Record"

9. The following IPMI commands or command groups are not yet implemented:

   • MicroTCA Commands

10. No XT1L temperature thresholds are used when an AM5901 hardware is detected.

11. Local MCH temperature is not used for temperature control. Only temperature events from AMC modules will be processed.

12. Common Option's bit 20 (processPwrShutdownEvent) will be set as a default when the MCH version R27 selects any configuration automatically. When bit 20 is set, some events coming from AMC boards will be ignored (filtered) in distinct M states to avoid e.g. unwanted shut downs. This event filtering is needed to run Kontron boards with not up-to-date IPMI firmware versions:

    • AM4001: IPMI version < 18hex
    • AM4002: IPMI version < 18hex
    • AM4010: IPMI version < 14hex
    • AM4011: IPMI version < 12hex
    • AM4100: IPMI version < 12hex

- AM4101: IPMI version < 11hex
- AM5010: IPMI version < 15hex

It is suggested to use only up-to-date IPMI versions and to set bit 20. In that case special Hot Swap events ("Backend Power Failure" and "Backend Power Shut Down") will be processed at any time as well. From MCH version R28 on bit 20 will be set as a default.

Please note: In R27 the description of the function of bit 20 was incorrect.

11.2   R28

1. The MCH can only be used in an OM6062 and ELMA K1 and K2 and OM6060 compatible chassis. I.e. it can't control MTCA specification conformant intelligent and redundant power units and cooling units.

2. Now the FCLKA is handled in the systems mentioned above.

3. Intelligent standard power supplies and fan controllers at IPMB-0 still are not handled.

4. MCH firmware upgrade is only possible by serial interface using a "Flash Tool" (lpcFlash, Flash Magic). No firmware backup and roll back to the former firmware version after a failed upgrade is implemented.

5. No AMC.0 R2.0 compatible clock e-keying is implemented. In OM6060 where the FCLKA must be switched on individually for every slot this will be done whenever a PCI Express connection is installed during e-keying.

6. The following FRU data records are not yet processed:

   - "MicroTCA Power Module Capability Record".
   - "MicroTCA FAN Geography Record".

7. The following IPMI commands or command groups are not yet implemented:

8. MicroTCA Commands

9. Local MCH temperature is not used for temperature control. Only temperature events from AMC modules will be processed.

10. Common Option's bit 20 (processPwrShutdownEvent) will be set as a default when the MCH version R28 selects any configuration automatically. When bit 20 is not set, some events coming from AMC boards will not be processed (filtered) in distinct M states to avoid e.g. unwanted shut downs. This event filtering is needed to run Kontron boards with not up-to-date IPMI firmware versions:

    - AM4001: IPMI version < 18hex.
    - AM4002: IPMI version < 18hex.
    - AM4010: IPMI version < 14hex
    - AM4011: IPMI version < 12hex
    - AM4100: IPMI version < 12hex
    - AM4101: IPMI version < 11hex
    - AM5010: IPMI version < 15hex

It is suggested to use only up-to-date IPMI versions and to set bit 20. In that case special Hot Swap events ("Backend Power Failure" and "Backend Power Shut Down") will be processed at any time as well. This behavior is e.g. needed to invoke a module shut down after a simple payload OS shutdown. When the module's MMC detects a manual OS shutdown the "Backend Power Shut Down" will be simulated to force a module shutdown.

11. E-keying for SATA port 3 in an OM6060 system fails because of an error in the match finder. A solution should be available with R29.

11.3   R29

1. The firmware now comes in three variants. They all are named "…AM4901…" but are still common for both the AM4901 and the AM5901:

    • IPMI_AM4901_FULL_Rxx.bin. This is the full firmware including boot block and application of version Rxx. It must be initially programmed using, for example, the tool 'lpcFlash' at the serial management interface.

    • IPMI_AM4901_FWUM_Rxx.bin. This is the application firmware for upgrade of an already programmed firmware in binary format. The tool 'ipmitool fwum …' is suitable for upgrading this firmware.

    • IPMI_AM4901_FWUM_Rxx.hpm. This is the application firmware for upgrade of an already programmed firmware in HPM.1 format. The tool 'ipmitool hpm …' is suitable for upgrading this firmware.

2. Upgrade is now possible during full operation in the field. The communication for the upgrade may be done:

    • Via IPMB using a processor AMC with KCS interface in the rack
    • Via Ethernet using the management port

3. The Ethernet management port is now operable. This allows a higher level access to the IPMI system from a remote system e.g. using 'ipmitool'. This may avoid running 'ipmitool' on a processor AMC in the rack. The MAC address for the management port is automatically taken from the CM FRU data if it is defined there.

4. The CLI commands 'lan set 1 …' and 'lan print 1' allow the set up of all basic LAN parameters for the Ethernet management port. Settable parameters are MAC address, IP address, net mask and access state.

5. Now a non-Kontron uTCA standard rack will be managed if it includes or emulates a standard Power Module which is addressed by a single IPMB address. Please note that a standard Cooling Unit is not yet supported.

6. The CLI command 'showerror' now shows errors in detail.
7. Four slot racks are not supported.
8. When using a standard power module, the option "permanent power" is not supported (e.g. the cli command "pwr" is not supported).
9. After having changed the current mode (e.g. cli command "currentopts"), the MCH must be reset (cli command "rs") or the MCH must be deactivated and then reactivated.

11.4 R30

New:

1. The boot block has been modified. Therefore a simple upgrade of the firmware does not enable all improvements. The old bootblock did not allow to keep the CLI cable connected during boot. Therefore: Flash the full R30 firmware with the tool lpcFlash then upgrade R30 (MCH type AM5901 only) to have R30 in both firmware banks.
2. Support of VLAN for the ethernet management port.
3. SOL support for the CLI serial interface via the ethernet management port.
4. The Master Write-Read command is now supported in a unique way by the CLI using the 'tipmi ...' command and the standard IPMI command.

   Command parameter 'bus ID' is re-structured in difference to the former implementation:

   [7:4] channel number (refer to eI2CIF_Interface in i2cifBoard.h:

   = 0 = for bus with carrier's EEPROM etc.

   = 1 = for bus with FRU/NVParams etc. EEPROM etc.

   = 2 = for bus with Temperature sensors etc.

   = 3 = for Inter MCH IPMB bus

   = 4 = for IPMB-L

   = 5 = for IPMB-A

   = 6 = for IPMB-B

   Note: channels 3...6 are for master write-read only accessible in test mode. Write with this command is only supported for channels 1, 3.

   For switching the test mode on please refer to the IPMI command 'Manufacturing Test On' in ipmb.c/IpmManufacturingTestOn().

   [3:1] bus ID = 1

   [0] bus type = 1

5. Parameters of IPMI command "Manufacturing Test On":

   Request data:

   [1:4] password = 0xb4 0x90 0x91 0x8b

   [5] Mode = 0x80 = Board test. The bits in [6] are valid for the MCH.

   [6] Test option bits for to switch off disturbing activities:

   Bit 0 = 1 = Standard access to IPMBs is off

   Bit 1 = 1 = Standard access to the i2c temperature device is off

   Bit 2 = 1 = Standard sensor readings are off

   Bit 3 = 1 = No standard MCH device is active (temp. control, power control etc.)

   All bits = 0: All kinds of hardware access by the standard firmware is allowed, i.e. the test mode is completely disabled. Use this to disable the board test again.

6. Access of AMCs at the IPMB-L bus using 'ipmitool ... -b 7 ...' now is possible for IPMB addresses > 0x74.

7. Supports one standard power supply at IPMB-0 address 0xC2 when a special rack is detected (OM6120, OM6040, OM6063) or when a default/standard rack is recognized or when the environmental options are set accordingly.

8. Supports one standard cooling unit at IPMB-0 address 0xA8 when a special rack is detected (OM6120, OM6063) or when a default/standard rack is recognized or when the environmental options are set accordingly.

9. E-keying supports now all combinations of SATA ports 2 and 3. Formerly only port 2 has been supported.

10. Automatic e-keying for fclock is supported when a backplane slot to slot connection for the fclock is specified and both connected AMC modules support clock e-keying.

11. CLI command 'fclockopts' allows manual setting/simulation of missing / wrong / inconvenient backplane data for fclock e-keying support. This data allows the processing of the fclock e-keying by the firmware to support the automatic fclock sharing between AMC modules and/or the back plane.

12. IPMI OEM commands for get/set of the fclockopts is supplied.

13. In the MCH's common options there is a new bit (22) defined which suppresses the red LED for the case that an eventProcessingError (e.g. event log full) is detected.

14. OM6120 rack's fan control type has been corrected to the correct type 2.

15. Do not install a firmware of version R30 if the MCH runs in a rack of type OM6060 or OM6040D. This could cause continuous resets

Restrictions:

1. The following types of FRU data records are not yet processed:

   • MicroTCA FRU Information Partition Record

   • MicroTCA Carrier Manager IP Link Record

   • MicroTCA Carrier Information Record

   • MicroTCA Shelf Manager IP Link Record

   • MicroTCA Shelf Information Record

   • MicroTCA FAN Geography Record

   Please note that the contained information normally is not needed for the operation of this MCH.

2. Only one Power Unit is supported
3. Only one cooling unit is supported
4. No MCH redundancy is implemented.

11.5 R31

New:

1. Do not install a firmware of version R30 if the MCH runs in a rack of type OM6060 or OM6040D. This could cause continuous resets.
2. Before the installation of a firmware of version >= R31 in a rack of type OM6060 or OM6040D the Ethernet management port must be disabled by setting switch 2/1 (AM5901) alternatively 3/1 (AM4901) to position "ON". If the switch is set to "OFF" after the installation and the MCH resets continuously then please set the appropriate switch to position "ON" again to disable Ethernet at management port. The cause of the malfunction is that the MCH's hardware configuration has to be modified. Please contact Kontron for further assistance about the required modification (solder jumper).
3. In standard non-Kontron racks now a maximum of two standard cooling units are managed. These units are working in parallel i.e. both have the same fan level at a time. If only one unit is installed or working no error will be reported. Only if there is no working unit at all is the error is detected (CLI command showerror), and the health LED lights up red.

Restrictions:

1. The following types of FRU data records are not yet processed:

   • MicroTCA FRU Information Partition Record

   • MicroTCA Carrier Manager IP Link Record

   • MicroTCA Carrier Information Record

   • MicroTCA Shelf Manager IP Link Record

   • MicroTCA Shelf Information Record

   • MicroTCA FAN Geography Record

   Please note that the contained information normally is not needed for the operation of this MCH.

2. Only one Power Unit is supported
3. The input of a password at the CLI allows only 12 seconds waiting time per character. A violation of this limit will cause a Watchdog reset of the MCH which will cause a reset of the whole rack.