

T-EMU: GRLIB IrqMp Device Model Manual



Prepared

Mattias Holm
Technical Manager

Approved

Michela Alberti
General Manager

Checked

Dan Søren Nielsen
QA Manager



Record of Changes

Author	Description	Rev	Date
Mattias Holm	Auto gen tables	1.1	2016-05-12
Mattias Holm	Initial Version	1.0	2015-07-31

Table of Contents

1. Introduction	2
2. Configuration	2
3. Attributes	2
3.1. Properties	2
3.2. Interfaces	3
3.3. Ports	3
4. Limitations	3

1. Introduction

The IrqMp is part of the GRLIB device library from Gaisler. It is a multiprocessor capable interrupt controller.

The controller supports among things the routing of interrupts to different processor cores, and also broadcasted interrupts.

2. Configuration

<code>config.nCpu</code>	Number of processors supported.
<code>config.enExtIrq</code>	Enable extended IRQs.
<code>pnp.config</code>	Plug and play configuration word for APB plug-and-play.
<code>cpu</code>	Up to 16 CPUs supported. IfaceRef property should be connected to the different CPUs.

3. Attributes

3.1. Properties

Name	Type	Description
<code>broadcast</code>	<code>uint32_t</code>	
<code>config.enExtIrq</code>	<code>uint8_t</code>	
<code>config.logInterrupts</code>	<code>uint8_t</code>	



Name	Type	Description
config.nCpu	uint8_t	
config.traceReads	uint8_t	
config.traceWrites	uint8_t	
cpu	[16 x iref]	
extIntAck	[16 x uint32_t]	
force	[16 x uint32_t]	
irqClear	uint32_t	
irqCtrl	[16 x iref]	
irqForce0	uint32_t	
irqLevel	uint32_t	
irqPending	uint32_t	
mask	[16 x uint32_t]	
mpStatus	uint32_t	
object.timeSource	object	Time source object (a cpu or machine object)
pnnp.bar	uint32_t	
pnnp.config	uint32_t	

3.2. Interfaces

Name	Type	Description
ApbIface	ApbIface	
DeviceIface	DeviceIface	
IrqClientIface	IrqClientIface	uptree interrupt handlers (e.g. CPUs)
IrqIface	IrqCtrlIface	
MemAccessIface	MemAccessIface	
ResetIface	ResetIface	

3.3. Ports

Prop	Iface	Description
irqCtrl	IrqClientIface	irq port

4. Limitations

The following deviations from real hardware are known to exist with this model:



-
- Broadcasted interrupts are broadcasted at the current time to all CPUs, if it was triggered by a non-synchronised event, the interrupt is raised at different times on the different cores. Depending on the IRQ frequency and the configured quanta length, this may result in problems.