

MB86297A 'Carminé' Interrupts

© Fujitsu Microelectronics Europe GmbH

History

Date	Author	Version	Comment
04.06.2008	Anders Ramdahl	1.00	First release

Warranty and Disclaimer

To the maximum extent permitted by applicable law, Fujitsu Mikroelektronik GmbH restricts its warranties and its liability for **all products delivered free of charge** (eg. software include or header files, application examples, **Application Notes**, target boards, evaluation boards, engineering samples of IC's etc.), its performance and any consequential damages, on the use of the Product in accordance with (i) the terms of the License Agreement and the Sale and Purchase Agreement under which agreements the Product has been delivered, (ii) the technical descriptions and (iii) all accompanying written materials. In addition, to the maximum extent permitted by applicable law, Fujitsu Mikroelektronik GmbH disclaims all warranties and liabilities for the performance of the Product and any consequential damages in cases of unauthorised decompiling and/or reverse engineering and/or disassembling. **Note, all these products are intended and must only be used in an evaluation laboratory environment.**

1. Fujitsu Mikroelektronik GmbH warrants that the Product will perform substantially in accordance with the accompanying written materials for a period of 90 days from the date of receipt by the customer. Concerning the hardware components of the Product, Fujitsu Mikroelektronik GmbH warrants that the Product will be free from defects in material and workmanship under use and service as specified in the accompanying written materials for a duration of 1 year from the date of receipt by the customer.
2. Should a Product turn out to be defect, Fujitsu Mikroelektronik GmbH's entire liability and the customer's exclusive remedy shall be, at Fujitsu Mikroelektronik GmbH's sole discretion, either return of the purchase price and the license fee, or replacement of the Product or parts thereof, if the Product is returned to Fujitsu Mikroelektronik GmbH in original packing and without further defects resulting from the customer's use or the transport. However, this warranty is excluded if the defect has resulted from an accident not attributable to Fujitsu Mikroelektronik GmbH, or abuse or misapplication attributable to the customer or any other third party not relating to Fujitsu Mikroelektronik GmbH.
3. To the maximum extent permitted by applicable law Fujitsu Mikroelektronik GmbH disclaims all other warranties, whether expressed or implied, in particular, but not limited to, warranties of merchantability and fitness for a particular purpose for which the Product is not designated.
4. To the maximum extent permitted by applicable law, Fujitsu Mikroelektronik GmbH's and its suppliers' liability is restricted to intention and gross negligence.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES

To the maximum extent permitted by applicable law, in no event shall Fujitsu Mikroelektronik GmbH and its suppliers be liable for any damages whatsoever (including but without limitation, consequential and/or indirect damages for personal injury, assets of substantial value, loss of profits, interruption of business operation, loss of information, or any other monetary or pecuniary loss) arising from the use of the Product.

Should one of the above stipulations be or become invalid and/or unenforceable, the remaining stipulations shall stay in full effect.

0 Abstract

This document describes the interrupt circuitry of the MB86297A graphics controller and how to best handle interrupts in a robust fashion.

1 References

Referenced documents:

CHWRM Fujitsu Microelectronics
 MB86297A 'Carminé' Hardware Reference Manual
 rev. 1.27, 27 May 2008

All Fujitsu Microelectronics documents listed above are available on the Fujitsu Graphics Solutions website:

<http://www.fujitsu.com/emea/services/microelectronics/displaycontrollers/>

2 Interrupt structure

The MB86297A has a hierarchical interrupt structure.

Each unit of the MB86297A contains one or more interrupt status registers, each having its corresponding interrupt mask register. After the interrupt mask has been applied to the interrupt status, an interrupt request is forwarded to the global control unit.

The global control unit has its own set of interrupt status and mask registers reflecting which units are currently requesting service. After the interrupt mask has been applied to the interrupt status, the resulting interrupt is forwarded to the INTA signal of the PCI interface for further processing by the CPU.

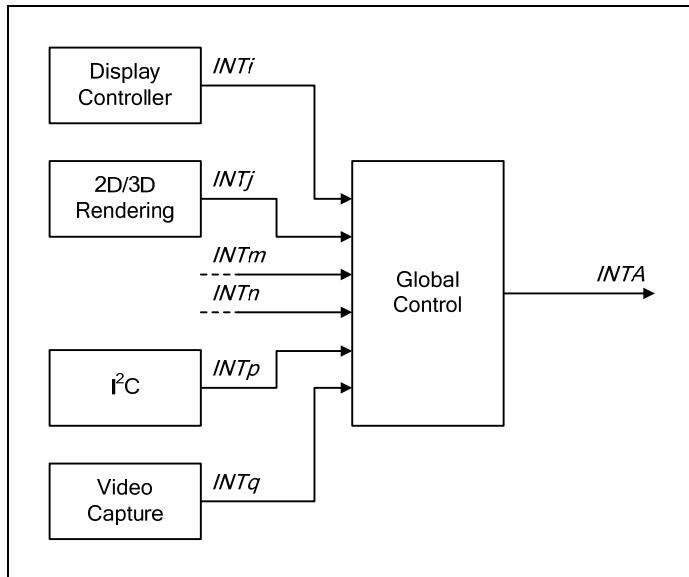


Figure 1 Interrupt Structure

The individual units request an interrupt by asserting (pulling high) its interrupt signal. The signal remains asserted until the interrupt is cleared, which is done by writing to the interrupt status register.

The global control unit sets an interrupt status flag when a rising edge is detected on the interrupt arriving from the corresponding unit. Once an interrupt flag has been set, it does not react on further interrupts from the unit until the flag has been cleared. The interrupts are cleared by writing one to the bits in the status register.

2.1 Clearing Interrupts

Contrary to the recommendations in the CHRWM, interrupt flags should be cleared in the global control unit before being cleared in the corresponding unit.



Clearing the interrupts in the order recommended in the CHWRM could lead to interrupts being lost; resulting in no further interrupts being serviced. This behavior has been observed for the capture unit VSYNC interrupt; see Appendix A for more details.

Appendix A Lost Capture Unit Interrupts

In one system it was observed that the interrupt from the capture unit got stuck after running the system for some time (1 to 20 minutes). When this occurred, the interrupt status of the capture unit was set but the corresponding status flag in the global control unit was not.

It was also observed that the problem was more likely to occur if the interrupt service routine was quickly clearing the interrupts. The interrupts were less likely to hang if a delay was added before clearing the interrupts.

Once the interrupt was hanging, it could be reactivated by toggling the interrupt mask of the capture unit. This would only be effective if the interrupt was already set. Toggling the interrupt mask would result in a rising edge being generated on the interrupt reaching the global control unit.

The interrupt service routine was clearing the interrupts in the order recommended in the CHWRM, i.e. the capture unit interrupt was cleared before the corresponding interrupt flag in the global control unit.

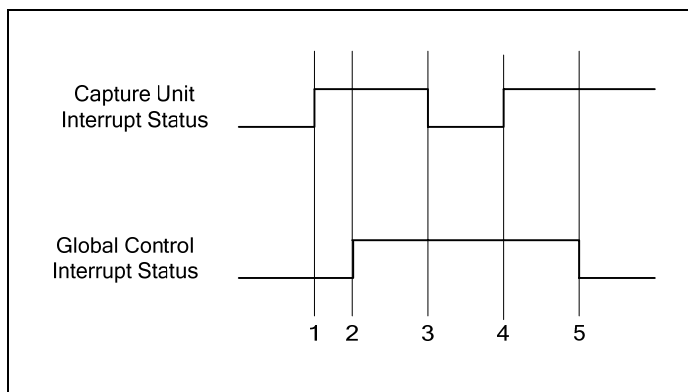


Figure 2 Hanging Capture Unit Interrupt

The only plausible explanation is that the capture unit VSYNC interrupt is issued more than once per frame, coupled with unfortunate timing of clearing the interrupts. Figure 2 shows this behavior. The different stages are explained below:

1. A VSYNC event occurs in the capture unit and the interrupt flag is set.
2. The global control detects the rising edge of the capture unit interrupt and sets the corresponding interrupt status flag. This interrupt request is forwarded to the CPU for further processing.
3. The interrupt service routine clears the VSYNC interrupt of the capture unit.
4. The VSYNC interrupt is reissued. Since the corresponding interrupt flag is already set in the global control unit, it does not detect the rising edge on the interrupt from the capture unit.
5. The capture interrupt flag of the global control unit is cleared by the interrupt service routine. Since the capture unit interrupt is now constantly set, no further rising edges will reach the global control unit. No further capture unit interrupts will be detected at this point.

The solution to this problem is to clear the interrupt flag in the global control unit before clearing the interrupt flag in the capture unit. This effectively prevents the deadlock situation described above from happening, since no rising edges on the interrupt line will be missed.

There is a small possibility that the interrupt is serviced more than once per frame. This is considered much less critical than the interrupt hanging.



It has not been possible to prove or disprove that the VSYNC interrupt is actually being issued more than once per frame. However, no other plausible explanation has been found.