5020 Brandin CT.
Fremont, CA94538,
USA
TEL: 1-510-683-3300 **FAX**: 1-510-683-3301

5th. Fl., No. 501-17
Chung-cheng Rd.,Hsin-tien
Taipei, Taiwan, R.O.C.
TEL: 886-2-218-5452 **FAX**: 886-2-218-5453

VT82C505 Interface with Other VL/ISA Chipsets

Date: August 10, 1994

1.0 Scope

This application note describes how the VT82C505 interfaces with 80486 and any core logic chipsets that support VL bus interface.

2.0 80486 and Chipset Interface

Most of the core chipsets available on the market that support VL and ISA bus use a 3 level subtracting decoding scheme to dispatch cycles among Local Memory, VESA devices or ISA devices. Cycles that first were not decoded as Local memory accesses, and, then not acknowledged by VESA devices become ISA cycles.

A VL to PCI bridge, such as the VT82C505, shall be treated as a VESA device on the VL local bus. It will acknowledge VL cycle initiator whenever the bus address falls within its decoding range, or a PCI device responds to that address on the VT82C505 secondary bus, the PCI bus.

3.0 VT82C505 Interface to VL Bus

The VT82C505 has both PCI and ISA programmable address ranges that can be used to determine and respond to VL cycle initiator timingly before the 2nd T2 by signal **LDEVO#**. According to the subtracting decoding scheme described above, whenever a cycle that does not fall within the VT82C505 programming address range and there is no other VL device claiming the cycle, the core chipset will always start an ISA cycle.

However, the size of programming register inside the VT82C505 is unlikely to fully reflect all the address space of PCI devices. In addition, a PCI device is only required to respond to the initiator between the 2nd and 5th cycle after **FRAME#** is asserted. Therefore, the dispatch decision could be delayed well after the 2nd T2 on the VL bus. In this case, to prevent an erratic ISA cycle being processed, a **Claim First Then Backoff** scheme is used to support the late acknowledgment problem.

3.1 Hardware Interface

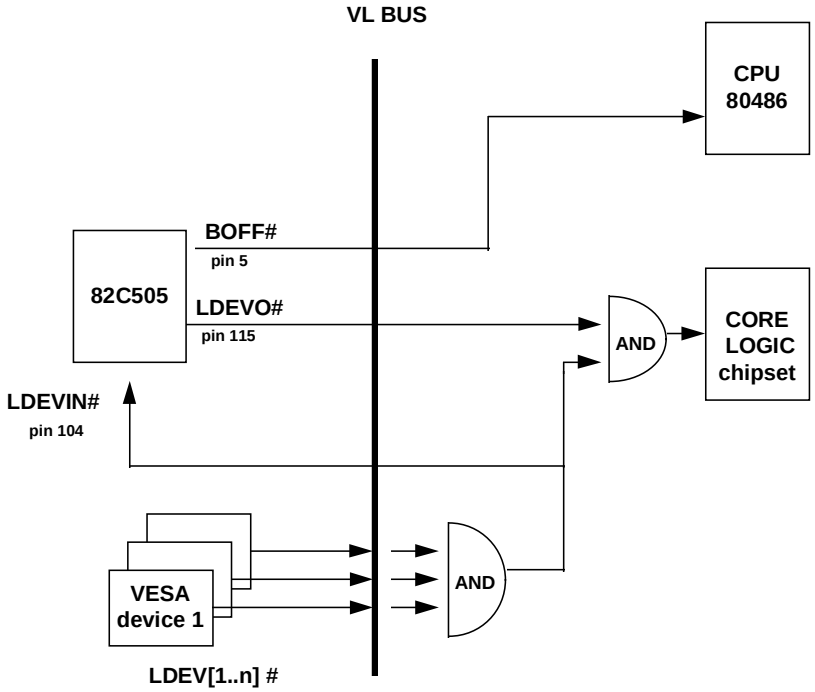
To interface to 80486 and the core logic chipset, the VT82C505 needs to monitor local device acknowledge signal from other VESA devices that input to **LDEVIN#** and generate its own acknowledge output, **LDEVO#** to VL bus controller of the Core chipset.

The VT82C505 also needs to generate **BOFF#** signal to abort a cycle when it applies **Claim First Then Backoff** operation.

The following diagram shows a rough sketch of how the signals should be connected.

The **pin 5** output from the VT82C505 is a multiple function pin that can be configured as **SERR#** or **BOFF#** depending on register bit value of RX96[3]. While this bit is programmed to be 1, the pin will be configured as **BOFF#**.

82C505 HARDWARE INTERFACE



3.2 Cycle and Timing Diagram

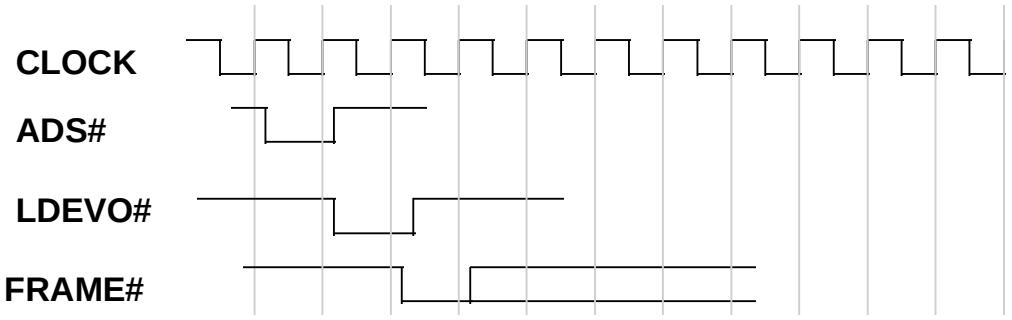
The VT82C505 dispatches PCI cycles depending on its programmed address range registers or Claim First Then BACKOFF operation. For those cycles address falling within the programmed address ranges, the bridge is able to make fast decision. Otherwise, the cycle dispatch mechanism needs to apply Claim First and Then Backoff operation that will produce a slower decision making process.

Fast Decoded PCI

The VT82C505 decoding logic uses PCI programmable register set, Rx87-89h, Rx8A-8Ch and Rx 8D-8Fh, along with a dynamic decoding algorithm to determine a match within PCI address range before the end of the 2nd T2. Once matched, the VT82C505 will claim the bus as a VESA device

and then generates a PCI cycle.

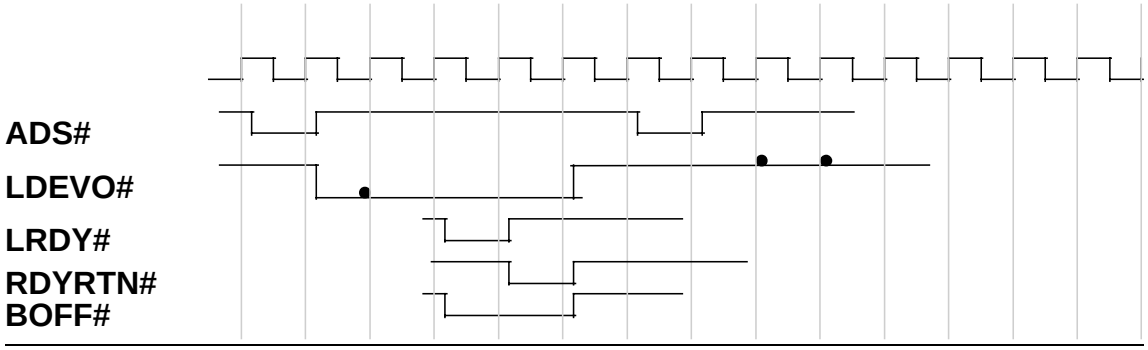
FAST PCI DECODE



Fast Decoded ISA

Similar to the PCI programming registers, Register Rx 92h along with IO address 00-FFh will be used by the decoding logic to determine whether ISA devises are being accessed or not before the end of the 2nd T2. Once an ISA address is matched, the VT82C505 will first claim VL cycle by asserting **LDEVO#** and then abort the cycle by asserting **BOFF#** signal at the 3rd T2. After the cycle is aborted, the CPU will restart the same cycle. However, the address space been accessed in the previous cycle is learned by the decoding logic so that the VT82C505 will not claim the cycle. The **LDEVO#** will not assert and the core chipset shall be able to generate an ISA cycle by rule of subtract decoding.

FAST ISA DECODE



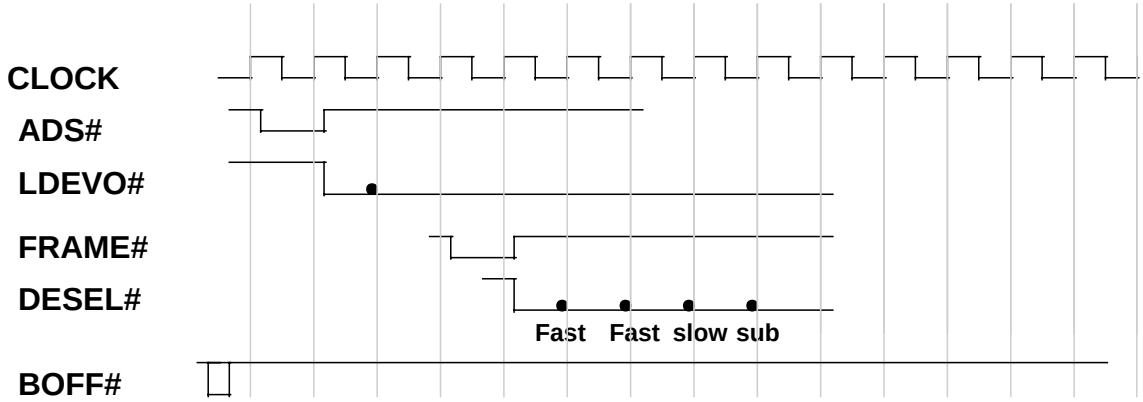
- Note that in Fast ISA decoded cycle no PCI cycle is generated.

Slow Decoded PCI

The PCI address accessed other than the programmed PCI windows and the dynamic decoding space will require **DEVSEL#** signal acknowledgment from the PCI bus. The **Claim First and Backoff** scheme will be utilized in this scenario.

Upon the cycle started, the VT82C505 will search through the PCI and ISA address map as well as the content of dynamic decode address.

SLOW PCI DECODE

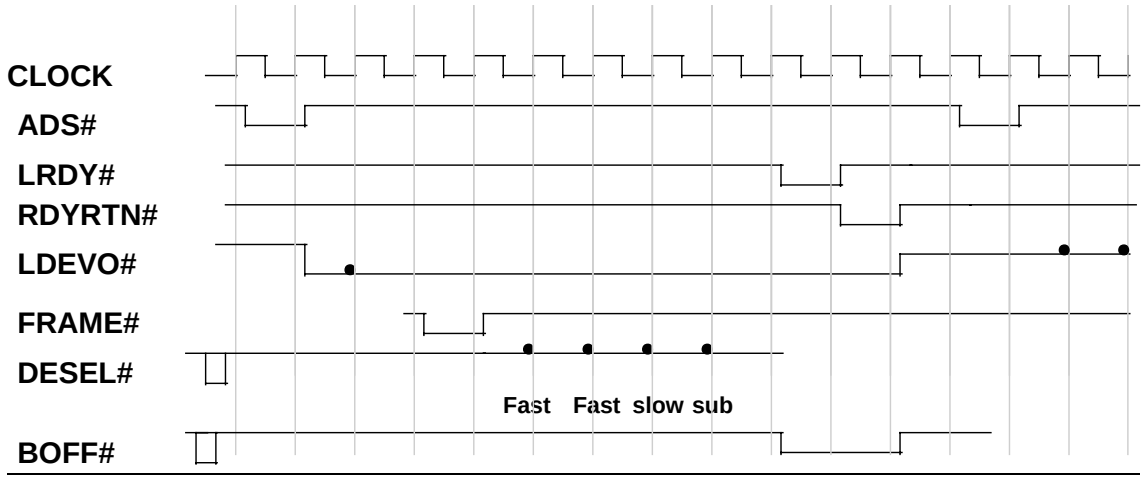


Slow Decoded ISA

If the ISA address does not fall within the programmed space or 00h to FFh IO address, the VT82C505 will start the **Claim First then Backoff** operation. Then, a PCI cycle will be initiated.. If no **DEVSEL#** responds within five clocks after **FRAME#** is asserted, the VT82C505 will consider it as an ISA cycle. The BOFF# will be asserted in the following clock to abort the current CPU cycle.

The CPU shall then terminate the cycle and regenerate it with the same address. Since the VT82C505 has learned the address as an ISA address, it will not claim the cycle. Therefore, the core chipset will start an ISA cycle.

SLOW ISA DECODE

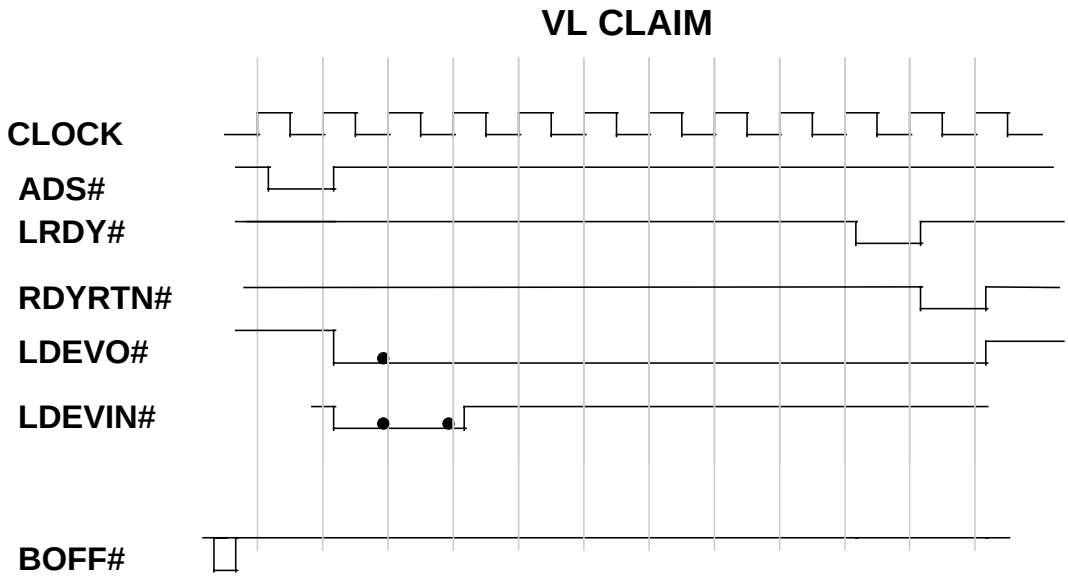


Without **LDEVO#** assert on the VL bus after the 2nd T2 in the regenerated CPU cycle, the core logic chipset will dispatch an ISA cycle according to the subtracting decoding rule mentioned in Sec. 2.0.

Claim by Other VESA Device

In case of other VESA devices claiming the bus, the initial state is similar to the slow ISA/PCI decode process.The VT82C505 will always claim the bus operation befroe the end of 2nd T2 due to the fact that the netither access address fall within the PCI or ISA programmed space, nor does it matches with any dynamic decoding address or learned ISA address.

In the mean time, if VT82C505 samples the validation of **LDEVIN#** from other VESA devises before the end of 2nd T2, it will recongnize this as a VL bus cycle instead of ISA or PCI cycle. The VT82C505 will abort all its internal operations at 3rd T2 with sole residual effect that **LDEVO#** remains being asserted until a valid **RDYRTN#** on VL bus is sampled.



4.0 BIOS Porting for VT82C505

. Enable Compatible mode access as early as possible

RX96<4> <- 1
RX96<3> <- 1

. Program ONBD DRAM configuration registers
(1. must be consistent with core chipset settings)
(2. must be programmed right after Enable Compatible mode)

Shadow registers	- RX30,RX31,RX32
384K relocation	- RX33
SMM	- RX5B
WRITE BACK	- RX 5E
ONBD DRAM size	- RX81

. Feature (performance option) settings (suggested user selectable items in BIOS setup)

CPU-PCI write buffer	RX82<7>
PCI-DRAM write buffer	RX82<6>
PCI-DRAM prefetch buffer	RX82<4>
Dynamic Decoding	RX82<3>
Burst write	RX82<1>
Burst read	RX82<0>
VL 0WS	RX83<7>
PCI Bus burst write enable	RX83<2>
*PCI Fast Back-to-Back	RX84<3>
Fast FRAME	RX84<2>

. Interrupt Handling

Conversion - global setting for INTA,B,C,D

RX86<5>	RX86<6>	
0	0	- Transparent (if PCI card using edge triggered)
0	1	- Transparent invert (if no PCI INT sharing allowed)
1	0	- Conversion (Intel compatible)
1	1	- Conversion (with EOI)

For VT82C505B, conversion mode is provided for individual INTA,B,C,D (RX97)

Steering - INT<A,B,C,D> can be steered to IRQ<5,9,10,11,14,15>, RX90,RX91

. Optional register for debugging purpose

PCI slave 1WS write	RX96<0>	
PCI master 1WS write	RX96<1>	
Retry timeout action	RX96<6>	
PCI Static Windows	RX87-8F	
Arbitration Mechanism	RX85	- for heavily-loaded system