1.0 Overview

Intelop's VME slave controller core is targeted for FPGAs or ASIC technology using their appropriate libraries during synthesis. The slave controller code is a parallel interface that is synchronous with the user side logic. The interface supports VME bus read write functions in various modes. When properly implemented in an FPGA or ASIC, it can be directly connected to the VME bus connector. All of the complex bus control functions are implemented inside the controller block and is very simple for user to interface to.

1.1Slave Controller

VME bus Slave interface simply monitors the Address and Data bus for Reads or Writes sent to it. Once a correctly decoded address is received the Slave will either receive information for a Write, or output information onto the Data bus in the case of a Read. The bus Master continues to control the Data bus during either interface. A Slave may also generate Interrupts over any of 7 IRQ lines. The Interrupts are acknowledged by the bus Controller.

2.0 Interface Features:

- Data modes: D8, D16, D32, D32-BLT.
- Address modes: A16 or A24 or A32
- Read, write, read-modify-write cycles
- Programmable rescinding DTACK
- Configurable D8, D16 or D32 interrupt logic
- User Selectable little/big endian conversion at the user interface
- Full synchronous user side interface for registers, peripherals and memories
- User selectable wait states

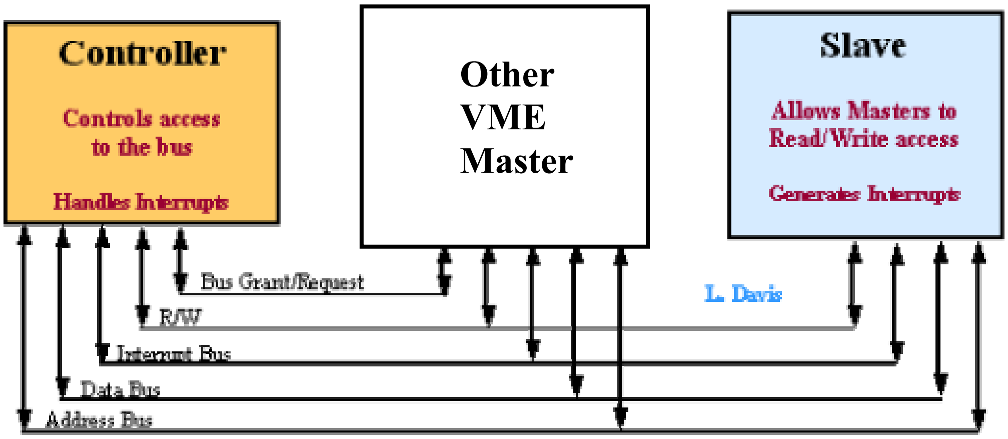
- It can operate at various clk frequencies up to 40 MHz in a Xilinx or Altera FPGA device for later interface enhancements

It adheres to IEEE 1014-1987{VME} The original VME Spec. that call for interfacing to signals from a 3 row P1/P2, 32 bit Data Xfers @ 40MBytes/second or higher for future upgrades.

The Data input has FIFO depth or 4 or 8 words for handling burst writes from the master. This provides much higher system performance
The data out has FIFO depth or 4 or 8 words for transferring burst writes to the mater. This provides much higher system performance

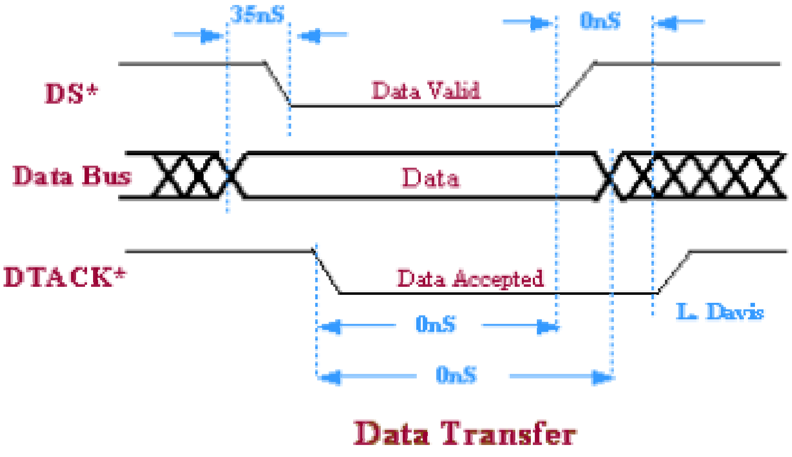
It offers full interrupt support.

During the start of a transfer, the Master will set the Data Transfer Bus [DTB] width using the two Data Strobes [DS0, DS1], Address bit 01 [A01] and bit 02 [A02], and LWORD. The condition of these lines at the start of a data transfer informs the Slave of the incoming data bus width. VME allowed 32 bit BLock Transfer [BLT]. VME64 added a 64bit Multiplexed BLock Transfer [MBLT] which uses the 32 bit data bus and the 32 bit address bus to transfer data.



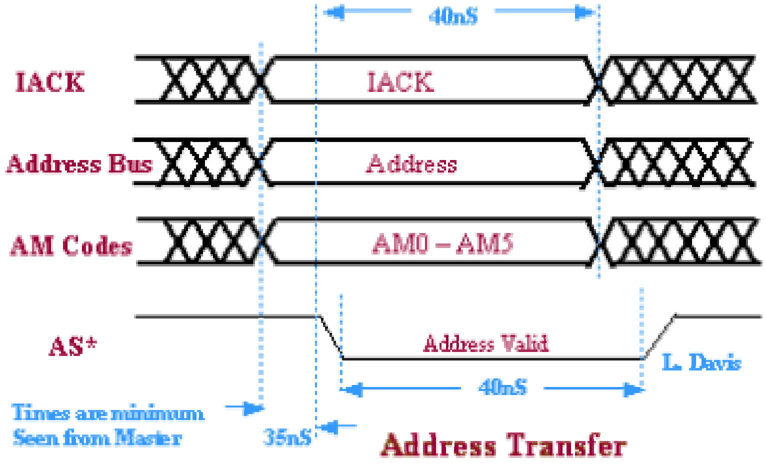
VMEbus System

VME Data Bus Transfer Timing

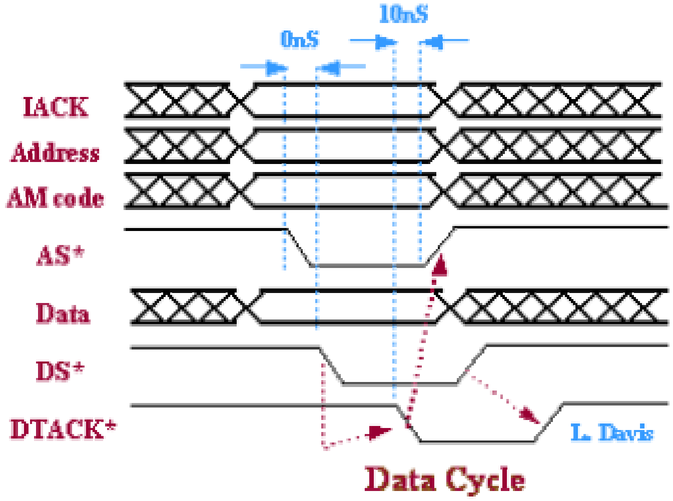


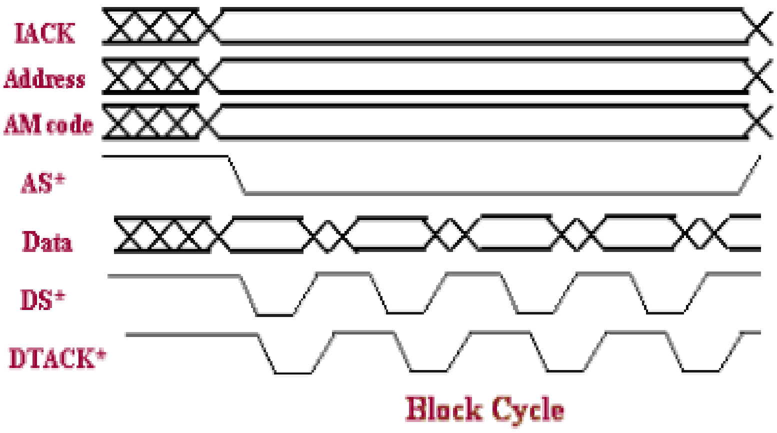
VME data bus Transfer timing: The Master places data on the Data Transfer Bus [DTB]. The Master then waits a minimum of 35nS before bringing one or both of the Data Strobe(s) [DS] low. The Data Strobe(s) going low indicate to the Slave that the Master has placed valid data on the bus. There is no defined time for the

Slave to acquire the data and acknowledge the transfer. How ever once the Slave has latched the data it will bring DTACK low. The Master will then release the Data Strobe(s). Once both of the data strobes are taken high the slave will release DTACK completing the data transfer cycle

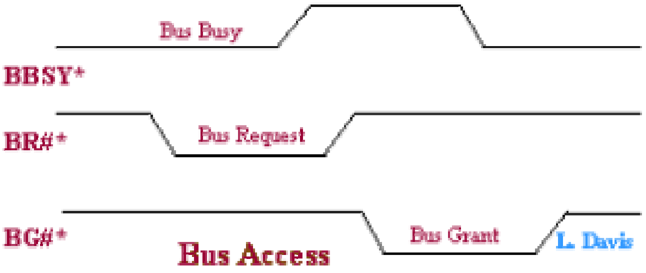


The VMEbus Master takes IACK high and places the address and AM [0-5] codes on the bus. Once the lines have been valid for 35nS the Master takes the Address Strobe [AS] low to indicate a valid address on the bus. For Interrupt cycles the IACK lines are driven low





A VMEbus BLock Transfer [BLT] consists of a single Address cycle followed by up to a 256 Byte Data transfer [in ether 8, 16, or 32 bit segments]. VME64 added the Multiplexed BLock Transfer [MBLT]. MBLT uses all 32 data bits and all 32 address bits to transfer 64 data bits at once over the bus



I/O description

The following sections describe the input and output ports of the VME core and provide an overview of their functionality.

3.1 General inputs

These pins are used to clock and initialize the whole VME core. To guarantee VME compliance, the falling edge of V_as* is used to latch the V_Adr and the V_am signals. The falling edge of the system clock 'clk' is used to guarantee interface timing on some signals. All other registers use the rising edge of 'clk' as the system clock.

pin name type description

clk in System clock

reset* in Asynchronous system reset, active low

3.2 VME Bus

These pins are used to control data transfer through the VME interface.

Pin name type description

V_Adr_in = [31:1] in VME address bus input

V_Adr_out = [31:1] out VME address bus output, used for MBLT

V_am[5:0] in = VME address modifier input

V_data_in[31:0] = in VME data bus input (from external bus driver)

V_data_out [31:0] = out VME data bus output (goes to external bus driver)

V_ext_ddrv* out = Active low drive enable signal for external bidirectional data bus drivers.

V_ext_ddir out = Direction control signal for external bidirectional data bus drivers:
'1' to VME bus. '0' from VME bus

V_ext_adrv* = out Active low drive enable signal for external bidirectional address/lword drivers.

V_ext_adir out Direction control signal for external bidirectional address/lword drivers:
'1' to VME bus. '0' from VME bus

V_int_ddrv* = in Active low drive enable signal for internal bidirectional data bus drivers.

V_int_adrv* in Active low drive enable signal for internal bidirectional address/lword drivers.
V_lword*_in in VME long word access indicator, low active
V_lword*_out out VME long word access indicator output, used for MBLT
V_dtack out Data transfer acknowledge. Used to indicate whether the DTACK is drive low or high (for rescinding)

3.3 Configuration

The core is configurable and can be optimized for a particular application. Prior to synthesis, these values should be fixed to select a desired configuration.

Pin name type description
Rescinding DTACK=
'0': Disabled
'1': Enabled

Config. interrupter in Interrupter selection
-8: D08(O) type interrupter
-16: D16 type interrupter
-32: D32 type interrupter
Config. endian in Endian selection for user side interface
‘0’: Big endian, transparent
‘1’: Little endian

3.4 User Side Interface

This part describes the user side interface, which is designed for easy and painless register and memory access.

3.4.1 Signal description

The entire user side interface is part of the clk system-clock domain. The user address modifier/decoder logic is not integrated in the core. This allows the user to build his own address decoding logic without changing the code of the VME. For convenience, more signals may be provided than a minimal approach would require. This simplifies the design of the user side logic.

Pin name/type description
int_user_addr(31:1) out Registered VME address bus (by falling edge of V_as*)
int_user_am(5:0) out Registered VME address bus modifier (by falling edge of V_as*)

user_access.mblt in MBLT user access signal. The user has 1-2 clock time for decoding of the address and address modifier and to assert the user_access_mblt signal when MBLT is addressed.
user_acc_rdy = User side acknowledgement signal. Active one event which finishes user side access.

user_addr(31:1) = out Registered VME address bus

user_am(5:0) = out Registered VME address bus modifier

user_wr_data(31:0) = out Write data bus.

user_rd_data(31:0) = in Read data bus. Must be valid when user_acc_rdy is 1.

user_rw* out = Data read/write_not signal.
‘1’: read data
‘0’: write data

user_byte_valid(3:0) out User data byte valid signal. Indicates which byte of the user_wr_data/user_rd_data bus are valid or requested.

Bit (0): user_wr_data(7:0) is valid

Bit (1): user_wr_data(15:8) is valid

Bit (2): user_wr_data(23:16) is valid

Bit (3): user_wr_data(31:24) is valid

user_ireq in Interrupt request. Active one indicates that an interrupt is pending and a VME interrupt will be generated. Must return to zero with user_iack = 1. user_iack out Interrupt acknowledgement. An active one event indicates the end of a valid interrupt acknowledge cycle.

core responds as D08(O), D16 or D32 interrupter.

Deliverables:

- Verilog source code or NetList.
- Verilog models for various components e.g. Memory/interface etc
- Verification suite
- Verilog Test bench

Pin name type description

VME Bus PinOut, P1 DIN Connector			
Pin #	Signal Name	Signal Name	Signal Name
	Row A	Row B	Row C
1	D00	BBSY*	D08
2	D01	BCLR*	D09
3	D02	ACFAIL*	D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK	BG3IN*	SYSFAIL*
11	GND	BG3OUT*	BERR*
12	DS1*	BRO*	SYSRESET*
13	DS0*	BR1*	LWORD*
14	WRITE*	BR2*	AM5
15	GND	BR3*	A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20	IACK*	GND	A18
21	IACKIN*	SERCLK	A17
22	IACKOUT*	SERDAT*	A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	IRQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31	-12v	+5v Standby	+12v
32	+5v	+5v	+5v