



# Macrocad Development Inc. IEEE 1284 Design Verification Models

## Description:

Macrocad's VSM 1284 product contains the models needed for design verification of a IEEE1284 bus system. These models act as IEEE1284 devices. They log transactions, execute negotiation, termination and transfers. They also check timing and signaling sequences. System designers can verify the correct implementation of their IEEE1284 host by simulating data transaction, and negotiation sequences with these models.

## Features:

- ◆ IEEE1284 spec compliant
- ◆ ECP, EPP 1.7, EPP 1.9, SPP, PS2, Byte, Nibble, and FIFOed SPP modes supported
- ◆ Decompression supported
- ◆ 1284 port time out features
- ◆ Burst sequences supported
- ◆ Automatic data checking
- ◆ Automatic timing checks
- ◆ IEEE1284 protocol checks
- ◆ Available in Verilog and VHDL versions

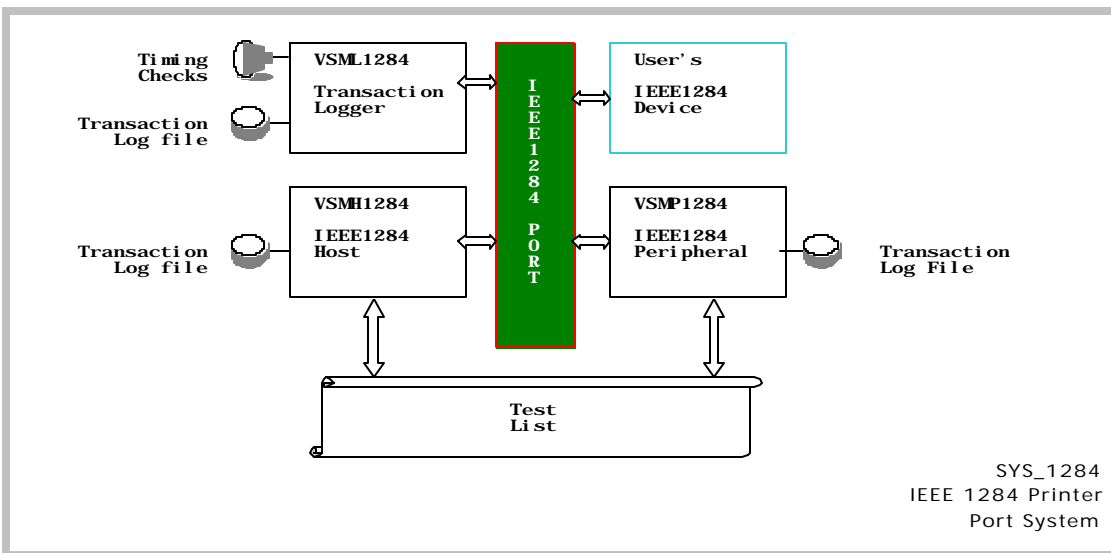


Figure 1



## Features

The VSM1284 model package is comprised of the sys\_1284, vsmh1284, vsmp1284 and vsml1284 models. The sys\_1284 model is the system level interconnect, and test list model. The vsmh1284 is the IEEE1284 host model. The vsmp1284 is the IEEE1284 peripheral model. The vsml1284 is the IEEE1284 logger model. These models can be used to construct a system which includes the IEEE1284 bus. It also provides the testing, timing and protocol checks for the IEEE1284 bus.

### VSMH1284 HOST MODEL

This model supports all the selection, negotiation, termination, data transfer, decompression, nibble, byte, error, and status modes set forth in the IEEE1284 specifications. The particular response of the model is determined by operating parameters. Tasks for each of the desired transactions are called, and options are passed to control the characteristics of the transaction.

Host task calls include:

initialize	this task initializes the model as it relates to the port signals
fw_spp	This task will execute an spp forward mode transaction
negotiate	this task will start and execute a negotiation sequence
terminate	this task will execute a termination sequence
rv_nab	This task will execute a reverse mode nibble or byte transaction
rv_ecp	This task will execute an ecp reverse mode transaction
fw_ecp	This task will execute an ecp forward mode transaction
rv_2_fw_ecp	This task will change from ecp reverse mode to ecp forward mode
fw_2_rv_ecp	This task will change from ecp forward mode to ecp reverse mode
manual_op	This task will execute a low level modification of the IEEE1284 signal states. The proper series of these task calls can result in a transaction, or error, or any other state for the IEEE1284 port.

The user can call the higher level task calls, and pass options to determine the type of characteristics for the transaction (ie. Host busy data not available, etc.) or construct a specific test from calling a sequence of manual\_op tasks. The higher level task calls are actually made up of a sequence of the manual\_op task calls. The manual\_op task can assert signals on the port, and wait for certain port signaling handshakes to occur.

### DESCRIPTION

This model operates and behaves like an IEEE1284 host device in the simulation environment. It supports Negotiation, Standard, Nibble, Byte, EPP, and ECP operations.

The vsmh1284 model operates through a transaction interface. The transactions are comprised of 64 bits of address, 64 bits of data and 64 bits of control and status. Transactions are automatically formed from the instruction in the test list.

Below is a list of the instruction types available.



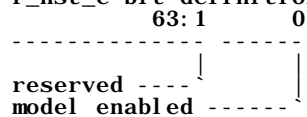
INSTRUCTION	DESCRIPTION
p_reset	resets the model
p_get_sbr	reads the selected sideband register
p_set_sbr	writes the selected sideband register
p_manual_op	manipulates individual IEEE1284 signals
p_negotiate	negotiate to selected mode
p_terminate	terminate back to SPP mode
p_fw_spp	forward SPP transfer
p_rv_nab	reverse NIBBLE or BYTE mode transfer
p_rv_nab_hint	reverse NIBBLE or BYTE mode host interrupt
p_fw_2_rv_ecp	forward to reverse ECP mode
p_rv_2_fw_ecp	reverse to forward ECP mode
p_fw_ecp	forward ECP transfer
p_rv_ecp	reverse ECP transfer
p_fw_epp	forward EPP transfer

The sideband register access is a read or write access to one of the following sideband registers.

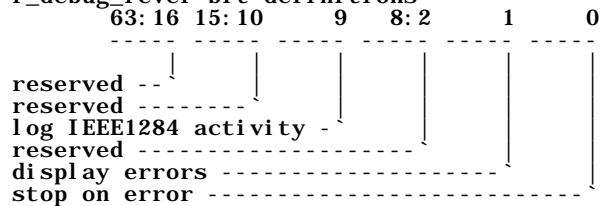
REGISTER	DESCRIPTION
r_hst_c	host options
r_debug_level	debug switches

The sideband registers` bit definitions are shown below.

**r\_hst\_c bit definitions**



**r\_debug\_level bit definitions**



The vsmh1284 host model is constantly tracking the state of the IEEE1284 bus. The s\_phase and s\_mode signals are used to track the changes, and can also be used to display the state on a simulator's waveform viewer. These signals are 32 bits, and contain the ASCII value of the IEEE1284 bus state. The ASCII values for the states are shown in the vsm1284 bus logger section.

**VSMP1284 PERIPHERAL MODEL**

The VSMP1284 model acts as a IEEE1284 peripheral device. It will respond to negotiations, termination, forward and reverse transfers. Transactions can be logged to a file.



Peripheral task calls include:

initialize	this task initializes the model as it relates to the port signals
negotiate	this task will respond to a negotiation sequence
terminate	This task will respond to a termination sequence
manual_op	This task will execute a low level modification of the IEEE1284 signal states. The proper series of these task calls can result in a transaction, or error, or any other state for the IEEE1284 port.

The VSMP1284 peripheral acts in a more passive way to the stimulus of the IEEE1284 port signals. Transactions will occur in an automatic way, with data being echoed back to the host.

## DESCRIPTION

This model is a IEEE1284 port device. It operates and behaves like an IEEE1284 peripheral device in the simulation environment. It supports Negotiation, Standard, Nibble, Byte, EPP, and ECP operations.

The vsmp1284 model operates through a transaction interface. The transactions are comprised of 64 bits of address, 64 bits of data and 64 bits of control and status. Transactions are automatically formed from the instruction in the test list. Below is a list of the instruction types available.

INSTRUCTION	DESCRIPTION
p_reset	resets the model
p_get_sbr	reads the selected sideband register
p_set_sbr	writes the selected sideband register
p_manual_op	manipulates individual IEEE1284 signals
p_negotiate	negotiate to selected mode
p_terminate	terminate back to SPP mode
p_fw_spp	forward SPP transfer
p_rv_nab	reverse NIBBLE or BYTE mode transfer
p_fw_ecp	forward ECP transfer
p_rv_ecp	reverse ECP transfer
p_rv_epp	reverse EPP transfer

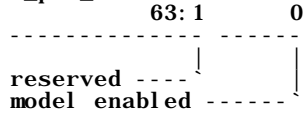
The peripheral model is more reactive than the host model due to the nature of the IEEE1284 bus. For instance, most IEEE1284 transfers in the peripheral model are done automatically. The reverse ECP and EPP tasks simply deposit data into a buffer for transfer at a time when the mode is negotiated to, and the host executed the reverse transfer. In this way, the peripheral reverse buffer can be loaded prior to any IEEE1284 negotiation or transfer. No interaction with the peripheral model is required to execute the reverse transfer. The host initiates the transfer, and the peripheral mode responds according to the data availability. The sideband register access is a read or write access to one of the following sideband registers.



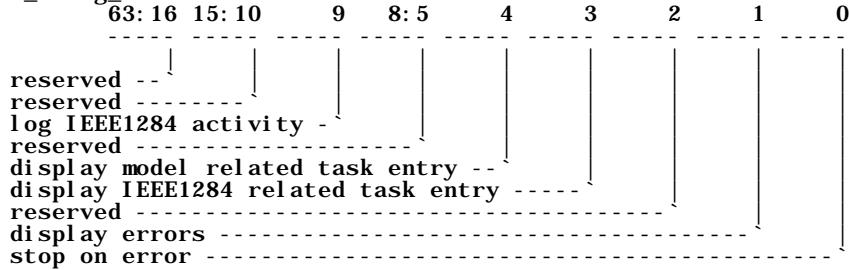
REGISTER	DESCRIPTION
r_per_c	peripheral options
r_debug_level	debug switches

The sideband registers` bit definitions are shown below.

r\_per\_c bit definitions



r\_debug\_level bit definitions



The vsmp1284 host model is constantly tracking the state of the IEEE1284 bus. The s\_phase and s\_mode signals are used to track the changes, and can also be used to display the state on a simulator's waveform viewer. These signals are 32 bits, and contain the ASCII value of the IEEE1284 bus state. The ASCII values for the states are shown in the vsml1284 bus logger section.

**VSML1284 TRANSACTION LOGGER**

This model logs all IEEE 1284 bus transactions to a file. This includes data transfers, (both byte and nibble) and negotiation sequences. This model does not participate in any IEEE 1284 transaction, but monitors all IEEE 1284 activity.

Parameters can be set to unique values for each instantiated model. These are to be considered constants, but some are used to set variables, which can be changed dynamically during simulation time. One such example is debug\_level. This is only used to load an initial value, (at the start of simulation time), into the debug register. This register can be dynamically changed during simulation, so display options can be switched on and off based on simulation time, sequential test instructions, IEEE 1284 status, error conditions, or other events.

The vsmp1284 peripheral model is constantly tracking the state of the IEEE1284 bus. The s\_phase and s\_mode signals are used to track the changes, and can also be used to display the state on a simulator's waveform viewer. These signals are 32 bits, and contain the ASCII value of the IEEE1284 bus state. The ASCII values for the states are shown below.



s_phase	DESCRIPTION
"fidl"	SPP mode forward idle
"fdat"	SPP mode forward data transfer
"rset"	SPP mode bus reset
"negt"	negotiation phase
"term"	termination phase
"ridl"	NAB mode reverse idle
"rdat"	NAB mode reverse data transfer
"hbnd"	NAB mode host busy, data not available
"hbda"	NAB mode host busy, data available
"inth"	NAB mode interrupt host
"setu"	ECP mode setup
"fidl"	ECP mode forward idle
"fdat"	ECP mode forward data transfer
"ridl"	ECP mode reverse idle
"rdat"	ECP mode reverse data transfer
"hrcv"	ECP mode host recovery
"ftor"	ECP mode forward to reverse
"rtof"	ECP mode reverse to forward
"idle"	EPP mode idle
"fawr"	EPP mode forward address write
"fard"	EPP mode forward address read
"fdwr"	EPP mode forward data write
"fdrd"	EPP mode forward data read
"inth"	EPP mode interrupt
"----"	unknown state

s_mode	DESCRIPTION
"spp "	SPP mode forward compatible mode
"ps2b"	NAB mode reverse byte mode
"ps2n"	NAB mode reverse nibble mode
"ecp "	ECP mode
"epp "	EPP mode

## SYS1284 SYSTEM MODEL

The sys\_1284 system model contains the interconnects between the various IEEE1284 models, and also contains the IEEE1284 tests.

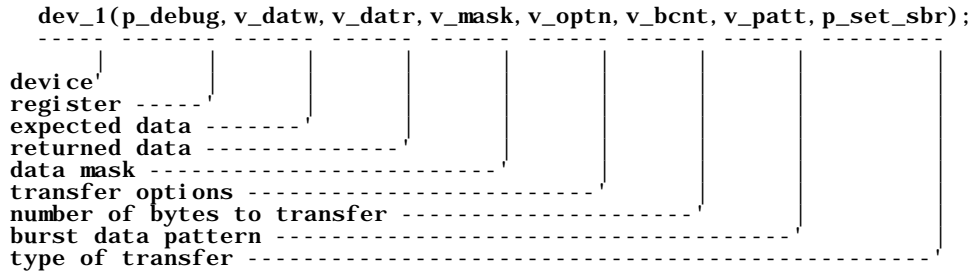
## DESCRIPTION

This model is the top level system model for the IEEE1284 design verification environment. It instantiates a logger and timing checker model, (vsml1284), a host model, (vsmh1284) and a peripheral model, (vsmp1284). Together, these model can create a robust IEEE1284 design verification environment. The tests communicate with these models via



a transaction bus. There are two signals for this bus, the mac\_req request signal, and the mac\_ret return signal. Tests are constructed of a list of various transaction instructions. The tests are executed depending of the bit values in the register r\_test\_mask. Each bit in this register determines if the corresponding test is executed. Thus test sequences can be controlled.

Each test is comprised of a list of test instructions. These instructions are automatically formed into transactions which handshake with the selected model. A typical test instruction is shown below. Below the test instruction is an explanation of the instruction fields.



The first 4 arguments are 64 bit vectors, and the next 4 are 8 bits. In the example above, the transaction is targeted to device 1. The register reference is to the debug register, (sideband register). The write or expected data is contained in the variable v\_datw. The returned data is deposited in the v\_datr variable. The v\_optn contains options, and the v\_bcnt the byte count, and v\_patt the data pattern. In the case of a sideband access, there are no options, data patterns or byte count arguments used. The last argument is the type of transaction requested, in this case a write access to a sideband register.

Below is a list of the instruction types available.

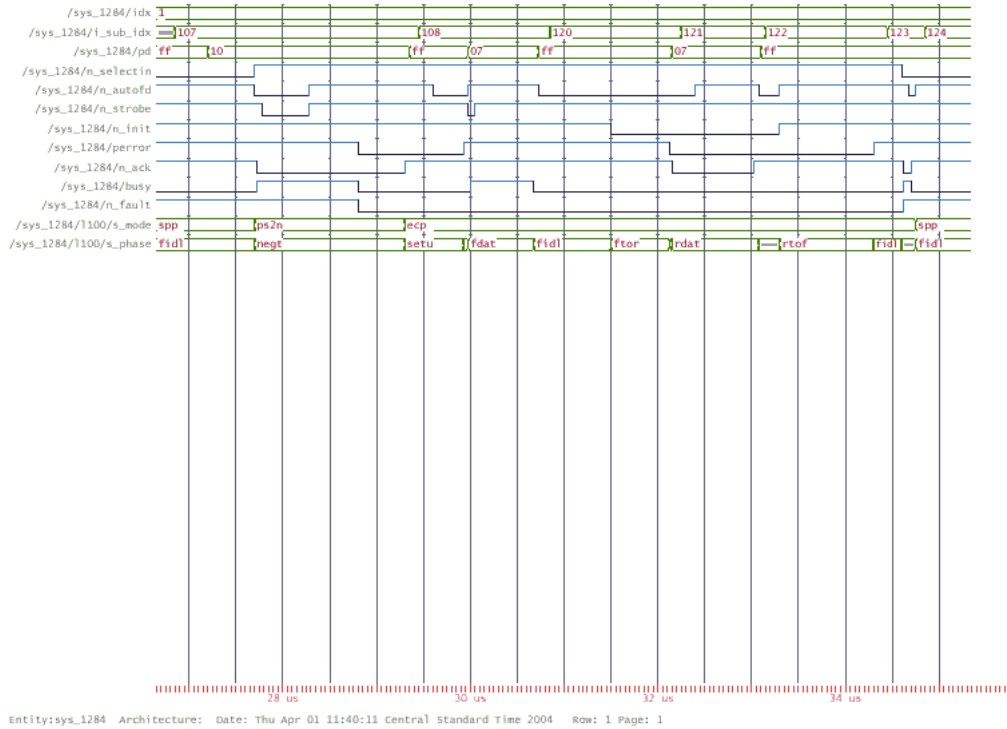
INSTRUCTION	DESCRIPTION
p_reset	resets the model
p_get_sbr	reads the selected sideband register
p_set_sbr	writes the selected sideband register
p_manual_op	manipulates individual IEEE1284 signals
p_negotiate	negotiate to selected mode
p_terminate	terminate back to SPP mode
p_fw_spp	forward SPP transfer
p_rv_nab	reverse NIBBLE or BYTE mode transfer
p_rv_nab_hint	reverse NIBBLE or BYTE mode host interrupt
p_fw_2_rv_ecp	forward to reverse ECP mode
p_rv_2_fw_ecp	reverse to forward ECP mode
p_fw_ecp	forward ECP transfer
p_rv_ecp	reverse ECP transfer
p_fw_epp	forward EPP transfer
p_rv_epp	reverse EPP transfer

The instructions are split into 2 basic types, sideband register access and IEEE1284 access. The sideband instructions are immediate, and data is not checked on sideband



register reads. The IEEE1284 instructions will execute the desired transfer, and return when the transfer is complete. Data can be checked on these instructions.

Below is a waveform of a typical negotiation, ECP transfer and termination.



**Reference Documents:**

DESCRIPTION	SOURCE	NAME
IEEE1284 specification	IEEE	1284-1994 IEEE Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers (ANSI)
ISA implementation	Microsoft, Hewlett Packard	"The IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard"