wait states used for access to each memory segment, controlling a system clock frequency, enabling parity checking functions, enabling direct memory access (DMA) functions, enabling random DMA requests, enabling random external interrupts, controlling cache burst refill sizes, selecting which cache burst word is to be retried, and controlling power supply voltage levels. It will be apparent that other parameters can be controlled or enabled depending upon the functions of a particular hardware circuit under design.

FIG. **3** illustrates a block schematic diagram of an exemplary hardware environment utilized to verify a design for an integrated circuit processor TR3100. The hardware environment illustrated in FIG. **3** provides capability to operate the integrated circuit processor TR3100 over its full range of functionality using the same diagnostic test suites developed for verifying a software model of the design for the integrated circuit processor TR3100.

The present invention has been described in terms of specific embodiments incorporating details to facilitate the understanding of the principles of construction and operation of the invention. Such reference herein to specific embodiments and details thereof is not intended to limit the scope of the claims appended hereto. It will be apparent to those skilled in the art that modifications may be made in the embodiment chosen for illustration without departing from the spirit and scope of the invention. Specifically, it will be apparent to one of ordinary skill in the art that the device of the present invention could be implemented in several different ways and the apparatus disclosed above is only illustrative of the preferred embodiment of the invention and is in no way a limitation.

What is claimed is:

**1**. An apparatus for verifying a design for a hardware circuit, the apparatus comprising:

a. a virtual system, wherein the virtual system comprises a host computer system programmed according to a first software program for causing the host computer to emulate the design for the hardware circuit;

b. a first test program for stimulating the virtual system to execute a function wherein the first test program comprises a first set up file and a first diagnostic test;

c. a hardware system, wherein the hardware system is functionally equivalent to the virtual system and wherein a parameter of the hardware system is selectively configured according to a parameter contained in the first set up file; and

d. a second test program for stimulating the hardware system to execute a function wherein the second test program comprises a second set up file and second diagnostic test and wherein the second diagnostic test is identical to the first diagnostic test.

**2**. The apparatus according to claim **1** wherein the virtual system is logically divisible into a software model of the design and into a virtual environment for the software model.

**3**. The apparatus according to claim **2** wherein the hardware system is physically divisible into a hardware circuit having equivalent functionality to the software model and into a hardware environment having equivalent functionality to the virtual environment.

**4**. The apparatus according to claim **1** wherein the parameter of the hardware system is selected from the group consisting of: assigning a memory segment for a particular function, selecting a number of wait states used for accessing a memory segment, controlling a system clock frequency, enabling a parity checking function, enabling a DMA function, enabling a DMA request, enabling a random external interrupt, controlling a cache burst refill size, selecting a cache burst word to be retried, and controlling a power supply voltage.

**5**. An apparatus for verifying a design for a hardware circuit, the apparatus comprising:

a. a software model for the design, wherein the software model is a first software program for causing a host computer to emulate the design;

b. a virtual environment wherein the virtual environment is a second software program for causing the host computer to emulate an environment for the design;

c. a first test program for stimulating the software model to execute a function within the virtual environment wherein the first test program comprises a first set up file and a first diagnostic test;

d. a hardware circuit for the design, wherein the hardware circuit is functionally equivalent to the software model;

e. a hardware environment operatively coupled to the hardware circuit, wherein the hardware environment is functionally equivalent to the virtual environment; and

f. a second test program for stimulating the hardware model to execute a function within the hardware environment wherein the second test program comprises a second set up file and second diagnostic test and wherein the second diagnostic test is identical to the first diagnostic test.

**6**. The apparatus according to claim **5** wherein the first set up file differs from the second set up file according to differences between initialization requirements of the virtual environment and initialization requirements of the hardware environment.

**7**. The apparatus according to claim **5** wherein the hardware environment is selectively configured according to a parameter contained in the first set up file.

**8**. A method of verifying a design for a hardware circuit, the method comprising steps of:

a. constructing a software model, wherein the software model is a first software algorithm which emulates the design for the hardware circuit;

b. constructing a virtual environment for interacting with the software model, wherein the virtual environment is a second software algorithm which emulates an environment for the design for the hardware circuit;

c. testing the software model by performing a diagnostic test on the software model;

d. constructing a hardware circuit which is functionally equivalent to the software model;

e. constructing a hardware environment operatively coupled to the hardware circuit wherein the hardware environment is functionally equivalent to the virtual environment; and

f. testing the hardware circuit by performing the diagnostic test on the hardware circuit.

**9**. The method according to claim **8** further comprising a step of integrating the hardware circuit into a single chip.

**10**. The method according to claim **8** further comprising steps of:

a. initializing the software model according to a first set up file before testing the software model; and

b. initializing the hardware circuit according to a second set up file before testing the hardware circuit.

**11**. The method according to claim **10** further comprising a step of configuring the hardware environment according to a predetermined parameters before testing the hardware circuit.

* * * * *