

CLAIMS

What is claimed is:

1. A method comprising:
 - modeling a system block of a system-on-chip (SoC) in a computer-implemented untimed functional model at an untimed functional level;
 - generating a computer-implemented transaction level (TL) model of the system block at a transaction level (TL) using the untimed functional model;
 - generating a computer-implemented cycle accurate (CA) model of the system block at a cycle accurate (CA) level using the TL model; and
 - verifying functionality of the SoC by operating a processor to execute a software program using at least one of the TL model and the CA model.
2. The method of claim 1, wherein the untimed functional model is implemented in a programming language selected from the group consisting of C, and C++.
3. The method of claim 1, wherein the TL model is a loosely timed (LT) model.
4. The method of claim 1, wherein the TL model is an approximately timed (AT) model.
5. The method of claim 1, wherein generating the TL model includes:
 - retrieving data from the untimed functional model;
 - encapsulating the data into a generic payload (GP); and
 - creating a transaction object from the generic payload at the transaction level.
6. The method of claim 1, wherein generating the TL model includes encapsulating the untimed functional model in a wrapper logic module configured to modify an interface between the untimed functional model and an external memory module.
7. The method of claim 6, wherein generating the CA model includes encapsulating the TL model in a transactor module configured to communicate with the wrapper logic module and a CA bus.

8. A method of transmitting data, the method comprising:
 - modeling at least one system block of a system-on-chip (SoC) at an untimed functional level in first and second untimed functional models;
 - generating first and second transaction level (TL) models of the at least one system block at a transaction level (TL) using the first and second untimed functional models, respectively;
 - generating first and second cycle accurate (CA) models at a cycle accurate (CA) level using the first and second TL models, respectively;
 - transmitting data from the first untimed functional model to the first CA model;
 - transmitting the data from the first CA model to the second CA model via a CA bus;
 - and
 - transmitting the data from the second CA model to the second untimed functional model.
9. The method of claim 8 wherein the first and second untimed functional models are implemented in a programming language selected from the group consisting of C and C++.
10. The method of claim 8, wherein generating the first TL model includes:
 - retrieving the data from the first untimed functional model;
 - encapsulating the data in a generic payload (GP); and
 - creating a transaction object from the generic payload.
11. The method of claim 10, wherein generating the first CA model includes encapsulating the first TL model in a first transactor module configured to communicate with the first TL module and the CA bus.
12. The method of claim 11, wherein transmitting data from the first untimed functional model to the first CA model includes:
 - sending the transaction object from the first TL module to the first transactor module; and
 - retrieving a payload comprising the data at the first transactor module.
13. The method of claim 12, wherein transmitting the data via the bus comprises driving the payload onto the CA bus.

14. The method of claim 13, wherein transmitting the data from the second CA model to the second untimed functional model includes:
 - retrieving the payload at a second transactor module of the second CA model;
 - sending the payload to the second TL model;
 - retrieving the data in the payload from a memory using an address specified by the payload; and
 - storing the data in the second untimed functional model.
15. A computer readable storage medium having instructions stored thereon, the instructions when executed by a processor causing the processor to perform the operations of:
 - modeling a system block of a system-on-chip (SoC) in an untimed functional model at an untimed functional level;
 - generating a transaction level (TL) model of the system block at a transaction level (TL) using the untimed functional model; and
 - generating a cycle accurate (CA) model of the system block at a cycle accurate (CA) level using the TL model.
16. The computer readable storage medium of claim 15, wherein the TL model is a loosely timed (LT) model.
17. The computer readable storage medium of claim 15, wherein the TL model is an approximately timed (AT) model.
18. The computer readable storage medium of claim 17, wherein generating the TL model includes:
 - retrieving data from the untimed functional model;
 - encapsulating the data into a generic payload (GP); and
 - creating a transaction object from the generic payload at the transaction level.
19. The computer readable storage medium of claim 17, wherein generating the TL model includes encapsulating the untimed functional model in a wrapper logic module configured to modify an interface between the untimed functional model and an external memory module.

20. The computer readable storage medium of claim 19, wherein generating the cycle accurate (CA) model includes encapsulating the TL model in a transactor module configured to communicate with the wrapper logic module and a CA bus.
21. A method comprising:
- generating a transaction level (TL) model of a system block of a system-on-chip (SoC) at a transaction level (TL);
 - generating an untimed functional model of the system block at an untimed functional level, the untimed functional model configured to be invoked by the TL model;
 - and
 - generating a cycle accurate (CA) model of the system block at a cycle accurate (CA) level using the TL model.

CLAIMS

1. A method comprising: modeling a system block of a system-on-chip (SoC) in a computer-implemented untimed functional model at an untimed functional level; generating a computer-implemented transaction level (TL) model of the system block at a transaction level (TL) by reusing the untimed functional model, wherein generating the TL model includes encapsulating the untimed functional model in a wrapper logic module configured to modify an interface between the untimed functional model and an external memory module; generating a computer-implemented cycle accurate (CA) model of the system block at a cycle accurate (CA) level by directly reusing the TL model, wherein the CA level has cycle accuracy at SoC signal level using signal-level timing granularity of the SoC, and wherein generating the CA model includes encapsulating the TL model in a transactor module configured to communicate with the wrapper logic module and a CA bus; and verifying functionality of the SoC by operating a processor to execute a software program using at least one of the TL model and the CA model.
2. The method of claim 1, wherein the untimed functional model is implemented in a programming language selected from the group consisting of C, and C++.
3. The method of claim 1, wherein the TL model is a loosely timed (LT) model.
4. The method of claim 1, wherein the TL model is an approximately timed (AT) model.
5. The method of claim 1, wherein generating the TL model includes: retrieving data from the untimed functional model; encapsulating the data into a generic payload (GP); and creating a transaction object from the generic payload at the transaction level.
6. A method of transmitting data, the method comprising: modeling at least one system block of a system-on-chip (SoC) at an untimed functional level in first and second untimed functional models; generating first and second transaction level (TL) models of the at least one system block at a transaction level (TL) by reusing the first and second untimed functional models, respectively; generating first and second cycle accurate (CA) models at a cycle accurate (CA) level by reusing the first and second TL models, respectively; transmitting data from the first untimed functional model to the first CA model; transmitting the data from the first CA model to the second CA model via a CA bus, wherein the CA bus is at SoC signal level and at a signal-level timing granularity of the SoC; and transmitting the data from the second CA model to the second untimed functional model.

7. The method of claim 6, wherein generating the first TL model includes: retrieving the data from the first untimed functional model; encapsulating the data in a generic payload (GP); and creating a transaction object from the generic payload.

8. The method of claim 7, wherein generating the first CA model includes encapsulating the first TL model in a first transactor module configured to communicate with the first TL module and the CA bus.

9. The method of claim 8, wherein transmitting data from the first untimed functional model to the first CA model includes: sending the transaction object from the first TL module to the first transactor module; and retrieving a payload comprising the data at the first transactor module.

10. The method of claim 9, wherein transmitting the data via the bus comprises driving the payload onto the CA bus.

11. The method of claim 10, wherein transmitting the data from the second CA model to the second untimed functional model includes: retrieving the payload at a second transactor module of the second CA model; sending the payload to the second TL model; retrieving the data in the payload from a memory using an address specified by the payload; and storing the data in the second untimed functional model.

12. The method of claim 6, wherein transmitting the data from the first untimed functional model to the first CA model includes transmitting the data from the first untimed functional model to the first TL model and transmitting the data from the first TL model to the first CA model, and wherein transmitting the data from the second CA model to the second untimed functional model includes transmitting the data from the second CA model to the second TL model and transmitting the data from the second TL model to the second untimed functional model.

13. A non-transitory computer readable storage medium having instructions stored thereon, the instructions when executed by a processor causing the processor to perform the operations of: modeling a system block of a system-on-chip (SoC) in an untimed functional model at an untimed functional level; generating a transaction level (TL) model of the system block at a transaction level (TL) by reusing the untimed functional model, wherein generating the TL model includes encapsulating the untimed functional model in a wrapper logic module configured to modify an interface between the untimed functional model and an external memory module; and generating a cycle accurate (CA) model of the system

block at a cycle accurate (CA) level by directly reusing the TL model, wherein the CA level has cycle accuracy at SoC signal level using signal-level timing granularity of the SoC, and wherein generating the cycle accurate (CA) model includes encapsulating the TL model in a transactor module configured to communicate with the wrapper logic module and a CA bus.

14. The computer readable storage medium of claim 13, wherein the TL model is a loosely timed (LT) model.

15. The computer readable storage medium of claim 13, wherein the TL model is an approximately timed (AT) model.

16. The computer readable storage medium of claim 15, wherein generating the TL model includes: retrieving data from the untimed functional model; encapsulating the data into a generic payload (GP); and creating a transaction object from the generic payload at the transaction level.