



(51) International Patent Classification:

G01R 31/317 (2006.01) G01R 31/3183 (2006.01)
G01R 31/3181 (2006.01) G01R 31/3187 (2006.01)

(21) International Application Number:

PCT/US2016/023866

(22) International Filing Date:

24 March 2016 (24.03.2016)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: INTEL CORPORATION [US/US]; 2200
Mission College Boulevard, Santa Clara, 95054 (US).

(72) Inventor: LI, Wei; 8124 Ladera Verde Dr., Austin, TX
78739 (US).

(74) Agent: IWASHKO, Lev; 1718 P St. NW, Unit 702,
Washington, DC 20036 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY,

BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM,
DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR,
KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG,
MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM,
PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC,
SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

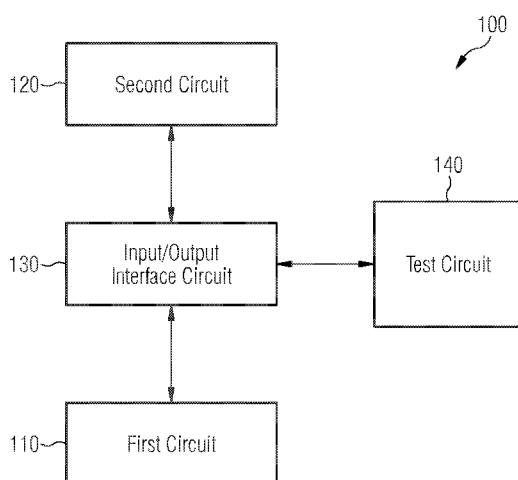
(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ,
TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU,
TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE,
DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU,
LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,
SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ,
GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: SYSTEM-ON-CHIP DEVICES AND METHODS FOR TESTING SYSTEM-ON-CHIP DEVICES

FIG. 1



(57) Abstract: A system-on-chip device includes a first circuit configured to operate in a first clock domain and a second circuit configured to operate in a second clock domain. A clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device. Further, the system-on-chip device includes an input/output interface circuit configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit and configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit. Additionally, the system-on-chip device includes a test circuit connected to the input/output interface circuit and configured to at least one of providing a test input signal to the input/output interface circuit and receiving a test output signal from the input/output interface circuit in a test mode of the system-on-chip device.

System-on-chip devices and methods for testing system-on-chip devices

Technical Field

5

The present disclosure relates to test concepts for electrical devices and in particular to system-on-chip devices and methods for testing system-on-chip devices.

10

Background

15

The integration of more and more complex circuits on a single semiconductor die or in a single package increases the complexity and efforts for testing the functionality of these circuits. Often various building blocks of different circuits are integrated as system-on-chip device. It is desired to improve the reliability of testing highly integrated circuits and/or decrease the test efforts and/or test time for highly integrated circuits.

Brief description of the Figures

20

Some examples of apparatuses and/or methods will be described in the following by way of example only, and with reference to the accompanying figures, in which

Fig. 1 shows a block diagram of a system-on-chip device;

25

Fig. 2 shows a block diagram of a part of a system-on-chip device with a test circuit using BILBO circuits;

Fig. 3 shows a BILBO circuit and a table of corresponding test modes;

30

Fig. 4 shows a schematic illustration of input and output channels between a CPU core circuit and a SOC uncore circuit;

Fig. 5 shows table of possible test modes of the test circuit of the system-on-chip device shown in Fig. 2;

Fig. 6A shows a block diagram of a part of a system-on-chip device illustrating connections to the input/output interface circuit from the CPU side;

Fig. 6B shows a block diagram of a part of a system-on-chip device illustrating connections to the input/output interface circuit from the SOC side;

Fig. 7 shows a block diagram of an input/output channel and a test circuit of a system-on-chip device in an outbound test mode;

Fig. 8 shows a block diagram of a part of a system-on-chip device illustrating a clock and voltage domain crossing;

Fig. 9 shows a block diagram of a part of a system-on-chip device;

Fig. 10 shows a block diagram of a system-on-chip device;

Fig. 11 shows a schematic illustration of a system-on-chip device;

Fig. 12 shows a flow chart of a method for a testing system-on-chip device; and

Fig. 13 shows a flow chart of another method for a testing system-on-chip device.

Detailed Description

Various examples will now be described more fully with reference to the accompanying drawings in which some examples are illustrated. In the figures, the thicknesses of lines, layers and/or regions may be exaggerated for clarity.

Accordingly, while examples are capable of various modifications and alternative forms, the illustrative examples in the figures and will herein be described in detail. It should be under-

stood, however, that there is no intent to limit examples to the particular forms disclosed, but on the contrary, examples are to cover all modifications, equivalents, and alternatives falling within the scope of the disclosure. Like numbers refer to like or similar elements throughout the description of the figures.

5

It will be understood that when an element is referred to as being “connected” or “coupled” to another element, it can be directly connected or coupled to the other element or intervening elements may be present. In contrast, when an element is referred to as being “directly connected” or “directly coupled” to another element, there are no intervening elements present. Other words used to describe the relationship between elements should be interpreted in a like fashion (e.g., “between” versus “directly between,” “adjacent” versus “directly adjacent,” etc.).

10

The terminology used herein is for the purpose of describing illustrative examples only and is not intended to be limiting. As used herein, the singular forms “a,” “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises,” “comprising,” “includes” and/or “including,” when used herein, specify the presence of stated features, integers, steps, operations, elements and/or component signals, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, component signals and/or groups thereof.

15

20

Unless otherwise defined, all terms (including technical and scientific terms) used herein have the same meaning as commonly understood by one of ordinary skill in the art to which examples belong. It will be further understood that terms, e.g., those defined in commonly used dictionaries, should be interpreted as having a meaning that is consistent with their meaning in the context of the relevant art and will not be interpreted in an idealized or overly formal sense unless expressly so defined herein.

25

Figure 1 shows a system-on-chip device 100 according to an example. The system-on-chip device 100 comprises a first circuit 110 configured to operate in a first clock domain and a second circuit 120 configured to operate in a second clock domain. A clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device 100. Further, the system-on-chip device 100

30

comprises an input/output interface circuit 130 configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit. Further, the input/output interface circuit 130 is configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit. Additionally, the system-on-chip device 100 comprises a test circuit 140 connected to the input/output interface circuit 130. The test circuit 140 is configured to provide a test input signal to the input/output interface circuit and/or receive a test output signal from the input/output interface circuit 130 in a test mode of the system-on-chip device.

- 10 By implementing a test circuit to provide test input signals to an I/O interface between circuits and/or to receive test output signals from the I/O interface, while the circuits are working at different speeds, the I/O interface can be tested under real operating conditions. In this way, the reliability of testing highly integrated circuits may be increased and/or the test efforts and/or test time for highly integrated circuits may be reduced, in comparison to other
- 15 test concepts.

The first circuit 110 may be any sub-circuit of a system-on-chip device 100 or any hardware building block of a system-on-chip device 100 connected to one or more other sub-circuits of the system-on-chip device 100 (at least the second circuit) or hardware building blocks of the system-on-chip device 100 through an on-chip input/output interface circuit 130 and operable in a clock domain, at least different from the clock domain of the second circuit 120. The first circuit 110 may be a core circuit (e.g. a CPU circuit), an uncore circuit (e.g. SOC circuit) or a memory circuit, for example. For example, the first circuit 130 is a digital circuit.

25

Similarly, the second circuit 120 may be any sub-circuit of a system-on-chip device 100 or any hardware building block of a system-on-chip device 100 connected to one or more other sub-circuits of the system-on-chip device 100 (at least the first circuit) or hardware building blocks of the system-on-chip device 100 through an on-chip input/output interface circuit 130 and operable in a clock domain, at least different from the clock domain of the first circuit 110. The second circuit 120 may be a core circuit (e.g. a CPU circuit), an uncore circuit (e.g. SOC circuit) or a memory circuit, for example. For example, the second circuit 120 is a digital circuit.

30

For example, circuits operable in different clock domains may be able to run at different speeds depending on a frequency of a clock signal provided to the different circuits. For example, a first clock signal having a first clock frequency is provided to circuits of the first clock domain (e.g. at least including the first circuit) and a second clock signal having a second clock frequency is provided to circuits of the second clock domain (e.g. at least including the second circuit). For example, the first clock frequency is different from (e.g. higher than) the second clock frequency at least in the first operating state of the system-on-chip device 100. For example, the first clock frequency may be higher than the second clock frequency in every operating state of the system-on-chip device 100. Alternatively or additionally, the first clock frequency may be equal to or lower than the second clock frequency in a second operating state of the system-on-chip device 100.

The input/output interface circuit 130 enables a data transfer between circuits running in different clock domains. For example, the input/output interface circuit 130 receives data as input signals (e.g. bit sequences carrying data to be transmitted) from the first circuit 110 with the clock frequency of the first clock domain and (e.g. at the same time) outputs data received from the first circuit 110 with the clock frequency of the second clock domain as output signals to the second circuit 120. Similarly, the input/output interface circuit 130 may receive data as input signals from the second circuit 120 with the clock frequency of the second clock domain and outputs the data received from the second circuit 120 with the clock frequency of the first clock domain as output signals to the first circuit 110. For example, the input/output interface circuit 130 is a digital circuit and the input signals and the output signals are digital signals. The input/output interface circuit 130 may comprise a memory circuit to store data provided by the first circuit 110 and/or the second circuit 120 to enable a balancing or compensation of the different clock speeds. For example, the first circuit 110 and the second circuit 120 exchange data through one or more input/output channels (e.g. request channel, response channel, data channel) of the input/output interface circuit 130. For example, the input/output interface circuit 130 may comprise a plurality of input/output channel circuits connected to the first circuit 110 and connected to the second circuit 120. For example, each input/output channel circuit of the plurality of input/output channel circuits may comprise a memory circuit configured to be written and to be read at different clock frequencies. Each memory circuit may comprise a first-in-first-out circuit (e.g. an asynchronous first-in first-out memory circuit or a bubble generation first-in first-out memory circuit).

The test circuit 140 provides a (digital) test input signal to the input/output interface circuit 130 and/or receives a (digital) test output signal from the input/output interface circuit 130 at least in the test mode of the system-on-chip device 100. For example, the test circuit 140 may provide a test input signal to the input/output interface circuit 130 instead of an input signal provided by the first circuit 110 to test a data transmission from the first circuit 110 to the second circuit 120 and/or may provide a test input signal to the input/output interface circuit 130 instead of an input signal provided by the second circuit 120 to test a data transmission from the second circuit 120 to the first circuit 110. Further, the test circuit 140 may receive a test output signal from the input/output interface circuit 130 caused by an input signal provided by the first circuit 110 or the test circuit 140 to test a data transmission from the first circuit 110 to the second circuit 120 and/or may receive a test output signal from the input/output interface circuit 130 caused by an input signal provided by the second circuit 120 or the test circuit 130 to test a data transmission from the second circuit 120 to the first circuit 110. The test circuit 140 may detect an erroneous behavior of the system-on-chip device 100 based on test output signals received from the input/output interface circuit 130 or may provide (e.g. directly or via a scan chain) the test output signal or a signal containing information on the test output signal to an externally accessible contact interface (e.g. pad) of the system-on-chip device 100 to enable an external detection of an erroneous behavior of the system-on-chip device 100.

The test circuit 140 may enable a test of the first circuit 110, the second circuit 120 and/or the input/output interface circuit 130 in the test mode of the system-on-chip device 100. Additionally, the test circuit 140 may enable a test of the first circuit 110, the second circuit 120 and/or the input/output interface circuit 130 in several different test modes of the system-on-chip device 100. For example, the test circuit 140 may provide the test input signal containing test input data at a write frequency (e.g. equal to the frequency of first or second clock domain). Additionally or alternatively, the test circuit 140 may read the test output signal at a read frequency (e.g. equal to the frequency of first or second clock domain). The write frequency may be higher than the read frequency in a first test mode and/or the write frequency may be lower than the read frequency in a second test mode. In this way, interfaces between circuits operating in some states faster than other circuits and in some states slower than the other circuits may be testable under real conditions, for example.

The test circuit 140 may be implemented in various ways and may be configured to support or perform one or more of different test scenarios. For example, the test circuit 140 is a digital circuit. For example, the test circuit 140 is configured or comprises one or more test sub-circuits being configured to operate in different modes. For example, the test circuit 140 may comprise at least one build-in logic block observer BILBO circuit (e.g. representing a test sub-circuit). The test circuit 140 may be configured to operate in one or more of the test modes described above and below while the system-on-chip device 100 is in a test mode.

For example, the test circuit 140 comprises a first test sub-circuit connected to the input/output interface circuit 130. The first test sub-circuit (or the test circuit) may be configured to provide a test input signal (e.g. a random or pseudo-random bit sequence) to the input/output interface circuit 130 in an input test mode of the first test sub-circuit (or the test circuit). For example, the first test sub-circuit may be a linear-feedback shift register LFSR or may be configured as linear-feedback shift register LFSR in the input test mode to generate a pseudo-random bit sequence as test input signal.

Additionally or alternatively, the first test sub-circuit (or the test circuit) may be configured to receive a test output signal from the input/output interface circuit 130 and provide an output test signal of the first test sub-circuit to an output (e.g. pad) of the system-on-chip device 100 or to a scan chain of the system-on-chip device 100 in an output test mode of the first test sub-circuit (or the test circuit). The output test signal of the first test sub-circuit may be equal to the test output signal received from the input/output interface circuit 130 or may be a signal containing information (e.g. signature) of the test output signal received from the input/output interface circuit 130. For example, the test sub-circuit may compress a continuous output test signal from the input/output interface circuit 130 into a signature indicating pass/fail of the input/output interface circuit 130. For example, the first test sub-circuit may be a multiple-input signature register circuit or may be configured as a multiple-input signature register MISR circuit configured to generate the output test signal of the first test sub-circuit indicating a signature of the test output signal of the input/output interface circuit 130.

Additionally or alternatively, the first test sub-circuit (or the test circuit) may comprise a shift register or may be configured as a shift register in a scan test mode of the first test sub-circuit (or the test circuit). The first test sub-circuit may be configured to connect the shift

register to a scan chain of the system-on-chip device 100 (e.g. a scan chain of the first circuit or the second circuit) in the scan test mode. In this way, the shift register of the first test sub-circuit may be testable by a scan chain test of the system-on-chip device 100 or data (e.g. the test output signal received from the input/output interface circuit or data received from the first circuit or the second circuit) captured by the shift register of the first test sub-circuit can be provided to the scan chain of the system-on-chip device 100. Additionally, the first test sub-circuit (or the test circuit) may be configured to capture output data of the first circuit 110, the second circuit 120 and/or the input/output interface circuit 130 by the shift register in a capture test mode of the first test sub-circuit (or the test circuit). For example, the first test sub-circuit may capture data launched by scan cells of the first circuit 110 and/or the second circuit 120.

Additionally or alternatively, the first test sub-circuit (or the test circuit) may be configured to provide predefined signals to the first circuit 110 and/or the second circuit 120 instead of the output signals of the input/output interface circuit 130 in a sealing test mode of the test circuit 140 during a test operation (e.g. scan test or ATPG test) of the first circuit 110 and/or the second circuit 120. In this way, a defined environment may be provided during the test of the first circuit 110 and/or the second circuit 120. For example, the predefined signals may be constant signals in the sealing test mode of the test circuit.

Optionally, the test circuit 140 may comprise a second test sub-circuit connected to the input/output interface circuit. For example, the second test sub-circuit may be configured to work in one or more test modes described in connection with the first test sub-circuit. For example, the second test sub-circuit may be configured to provide a test input signal to the input/output interface circuit 130 in an inbound test mode of the test circuit 140. Further, the first test sub-circuit may be configured to receive a test output signal from the input/output interface circuit based on the test input signal provided by the second test sub-circuit in the inbound test mode of the test circuit 140. In this way, a data transfer from the second circuit 120 to the first circuit 110 (or vice versa) through the input/output interface circuit 130 may be tested.

Optionally, the test circuit 140 may comprise a third test sub-circuit connected to the input/output interface circuit 130. For example, the third test sub-circuit may be configured to work in one or more test modes described in connection with the first test sub-circuit. For

example, the first test sub-circuit is configured to provide a test input signal to the input/output interface circuit 130 in an outbound test mode of the test circuit 140. Further, the third test sub-circuit may be configured to receive a test output signal from the input/output interface circuit 130 based on the test input signal provided by the first test sub-circuit in the outbound test mode of the test circuit 140. In this way, a data transfer from the first circuit 110 to the second circuit 120 (or vice versa) through the input/output interface circuit 130 may be tested.

Optionally, the second test sub-circuit may comprise a shift register or the second test sub-circuit may be configured as a shift register in a scan test mode of the test circuit. Further, the third test sub-circuit may comprise a shift register or the third test sub-circuit may be configured as a shift register in the scan test mode of the test circuit. For example, the second test sub-circuit may be configured to connect the shift register of the second test sub-circuit to a scan chain of the system-on-chip device 100 in the scan test mode of the test circuit and the third test sub-circuit may be configured to connect the shift register of the third test sub-circuit to the scan chain of the system-on-chip device 100 in the scan test mode of the test circuit.

For example, the first test sub-circuit may be placed or located between the first circuit 110 and the input/output interface circuit 130. The second test sub-circuit and/or the third test sub-circuit may be placed or located between the second circuit 120 and the input/output interface circuit 130. For example, the first test sub-circuit, the second test sub-circuit and/or the third test sub-circuit may comprise or may be a build-in logic block observer BILBO circuit (e.g. Fig. 3).

The system-on-chip device 100 may be an integrated circuit comprising a plurality of different hardware building blocks or circuits implemented on the same semiconductor die. For example, the first circuit 110, the second circuit 120, the input/output interface circuit 130 and the test circuit 140 of the system-on-chip device 100 are located on the same semiconductor die.

The following examples relate to a first circuit 110 being a core circuit (e.g a CPU) and a second circuit 120 being an uncore circuit (e.g. a SOC circuit), although the described test

concept may be applicable to a an input/output interface between other circuits of a system-on-chip device as well.

Fig. 2 shows a block diagram of a part of a system-on-chip device 200 according to an example. The system-on-chip device 200 comprises as a first circuit a central processing unit (CPU), not shown in Fig. 2, and as a second circuit an uncore circuit 210 (SOC circuit).

An input/output interface circuit, which can be employed to communicate signals between the CPU and the uncore circuit 210, for example, when the CPU and the uncore circuit operate with different clock frequencies as in the first operating state or in the second operating state of the system-on-chip device 200, comprises three CPU-to-uncore (C2U) channel circuits and three uncore-to-CPU (U2C) channel circuits.

The C2U channel circuits comprise a C2U request channel circuit 260, a C2U response channel circuit 262, and a C2U data channel circuit 264.

15

The U2C channel circuits comprise a U2C request channel circuit 250, a U2C response channel circuit 252, and a U2C data channel circuit 254.

Output signals of the CPU, which are designated for the uncore circuit 210, are transferred through at least one of the C2U channel circuits to the uncore circuit 210. Analogously, outputs signals of the uncore circuit 210, which are designated for the CPU, are transferred through at least one of the U2C channel circuits to the CPU.

In order to test at least one of the C2U channel circuits and the U2C channel circuits the system-on-chip device 200 comprises a test circuit, comprising a first test sub circuit, implemented as Internal BILBO network 240, a second test sub circuit, implemented as Inbound BILBO network 220, and a third test sub circuit, implemented as Outbound BILBO network 230.

30 In an outbound test mode of the test circuit, the Internal BILBO network 240 is configured to provide test input signals to at least one of the C2U channel circuits. A multiplexer 242 of the CPU can be employed to forward these test input signals to at least one of the C2U channel circuits under test, instead of the output signals of the CPU designated for the uncore circuit 210. Corresponding to the test input signals from the Internal BILBO network

240, the C2U channel circuits under test may provide test output signals at their outputs that are both connected to inputs of the Outbound BILBO network 230 as well as to the uncore circuit 210 via a set of outbound multiplexers 231, comprising a first outbound multiplexer 232, a second outbound multiplexer 234, a third outbound multiplexer 236, and a fourth
5 outbound multiplexer 238 (additional or less outbound multiplexers may be implemented according to the number of outputs of the C2U channel circuits). During the outbound test mode of the test circuit, the Outbound BILBO network 230 may thus receive the test output signals from the C2U channel circuits under test.

10 In an inbound test mode of the test circuit, the Inbound BILBO network 220 is configured to provide test input signals to at least one of the U2C channel circuits. A set of inbound multiplexers 221, comprising a first inbound multiplexer 222, a second inbound multiplexer 224, a third inbound multiplexer 226, and a fourth inbound multiplexer 228, can be employed to forward these test input signals to at least one of the U2C channel circuits, instead
15 of the output signals of the uncore circuit 210 designated for the CPU (additional or less inbound multiplexers may be implemented according to the number of inputs of the U2C channel circuits). Corresponding to the test input signals from the Inbound BILBO network 220, the U2C channel circuits under test may provide test output signals at their outputs that are both connected to inputs of the Internal BILBO network 240 as well as to the CPU.

20

Moreover, the outputs of the uncore circuit 210, which are designated for the CPU, are connected to data inputs of the Inbound BILBO network 220, so that it is possible to send signals from the uncore circuit 210 to the Inbound BILBO network 220.

25 Additionally, to send signals from the C2U channel circuit to the uncore circuit 210, data outputs of the Outbound BILBO network 230 are connected via the set of outbound multiplexers 231 to the uncore circuit 210. According to the setting of the set of outbound multiplexers 231, the uncore circuit 210 may hence either receive signals from the Outbound BILBO network 230 or from at least one of the C2U channel circuits.

30

The Inbound BILBO network 220 may furthermore be connected to the uncore circuit 210 (or the CPU) via a serial data input, labelled as "SI from SOC". A serial data output of the Inbound BILBO network 220 may be connected to a serial data input of the Outbound BILBO network 230 (not shown in Fig. 2). Further, a serial data output of the Outbound BILBO

network 230, labeled as “SO to SOC”, may be connected to the uncore circuit 210, the Inbound BILBO network 220 and the Outbound BILBO network 230 may be configured to exchange serially data with the uncore circuit 210 (or the CPU) and/or be incorporated into a scan chain of the uncore circuit 210 (or the CPU).

5

Further, the CPU comprises a control circuit 270 (CPU TAP) and a control register circuit 272 (BGF BIST Creg) configured to control the Inbound BILBO network 220, the Outbound BILBO network 230, and the Internal BIBLO network 240, the set of outbound multiplexers 231, the set of inbound multiplexers 221, and the multiplexer 242 of the CPU.

10

The CPU TAP 270 may send a “GNB_config[1:0]” control signal to a mode decoder circuit 284 of a test control circuit 280 of the test circuit. Further, the uncore circuit 210 may send a “scan_enable” control signal to the mode decoder circuit 284. The mode decoder circuit 284 may be configured to decode the “GNB_config[1:0]” control signal and the “scan_enable” control signal, and provide corresponding input control signals to the Inbound BILBO network 220, the Outbound BILBO network 230, and the Internal BIBLO network 240, to set them to different test modes (e.g. the inbound test mode or the outbound test mode).

15

In detail, the mode decoder circuit 284 provides the “internal B[2:1]” input control signal to the Internal BILBO network 240, the “inbound B[2:1]” input control signal to the Inbound BILBO network 220, and the “outbound B[2:1]” input control signal to the Outbound BILBO network 230.

20

The test control circuit 280 comprises a clock control circuit 286, which is controlled by the BGF BIST Creg 272, to provide different clock signals (e.g. clock signals differing in at least one of frequency or voltage) to the Inbound BILBO network 220, the Outbound BILBO network 230, and the Internal BIBLO network 240. The clock control circuit 286 receives the input clock signals “mclk”, which is a function clock of the CPU, “uclk”, which is a function clock of the uncore circuit, and “ssclk”, which can be a clock of the scan chain of the uncore circuit in either a scan mode, designated as “SOC scan shift clock”, or in a capture mode (e.g. “SOC scan capture clock”). In the capture mode the frequency of ssclk may be at-speed (e.g. equal to the frequency of uclk). In addition, the clock control circuit

25

30

286 may receive a clock input signal “tck” (not shown in Fig. 2), which is the clock of the CPU TAP 270 and a scan chain of the CPU.

The clock control circuit 286 may multiplex and gate its input clock signals to the Inbound
5 BILBO network 220, the Outbound BILBO network 230, and the Internal BIBLO network 240 corresponding to different test modes of the Inbound BILBO network 220, the Outbound BILBO network 230, and the Internal BIBLO network 240. The “Internal_clk” output clock signal may be provided to the Internal BILBO network 240, the “inbound_clk” output clock signal may be provided to the Inbound BILBO network 220, and the “outbound_clk” output clock signal may be provided to the Outbound BILBO network 230.
10

The test control circuit 280 comprises a counter circuit 282, which is controlled by the BGF BIST Creg 272 and configured to count clock cycles of a clock signal provided by the clock control circuit 286.
15

Subsequently, an example is described, which uses a BILBO as a basic element to construct a BILBO network, which may support four different test and debug usage scenarios of CPU and SOC interface.

20 An example for an overall architecture (BILBO network architecture) is illustrated in Figure 2. For example, it comprises four major building blocks. They are an inbound BILBO segment, an outbound BILBO segment, an internal BILBO segment and a BILBO control block.

25 Concerning the inbound BILBO segment and the outbound BILBO segment, the length of the inbound BILBO segment may be equal to the total length of all payload sub channels from three U2C channels. Each U2C channel may correspond to a dedicated inbound BILBO segment. The three segments may be stitched into one inbound BILBO segment. The outbound BILBO may be made of three segments, with each of them corresponding to the
30 payload of a C2U channel.

The inbound BILBO block may be used as an LFSR when running inbound BGF BIST for the U2C channels. The outbound BILBO segment may be used as a MISR when running outbound BGF BIST for the C2U channels. Inbound and outbound BILBO segments may

be stitched together and deployed in the CPU and SOC boundary. For example, they may be accessed through both CPU TAP and SOC scan system for debug and at-speed scan testing.

- 5 The internal BILBO segment may be a shared segment, whose length may be the bigger of inbound segment length and outbound segment length. Internal BILBO segment may be used as a MISR when running inbound BGF BIST for U2C channels and may be used as an LFSR, when running outbound BGF BIST for C2U channels. The internal BILBO may also be accessible through CPU TAP and used for debug.

10

The BILBO control block may, for example, take the inputs from the CPU TAP as well as CPU control registers (Cregs) and provide control signals and clocks to the inbound, outbound and internal BILBO segments according to different usage modes. The control signals comprise BILBO configuration (B1, B2) as shown in Figure 3 as well as the multiplex-
15 er (mux) selection signals in Figure 2. The clocks for the three BILBO segments may be generated in this block according to different modes. Clock muxing and gating may be handled in this block.

- For example, in normal function mode (e.g. first and/or second operating mode of the system-on-chip device), all of 3 test sub-circuits (e.g. internal BILBO, inbound BILBO and
20 outbound BILBO) may be clock gated, so that they do not run in functional mode.

- In outbound test mode, only the first test sub-circuit (e.g. used as transmitter) and the third test sub-circuit (e.g. used as receiver/MISR) may be used and the second test sub-circuit is
25 shut off, for example.

- In inbound test mode, only the second test sub-circuit (e.g. used as transmitter) and the first test sub-circuit (e.g. used as receiver/MISR) may be used and the third test sub-circuit is shut off, for example.

30

In SoC scan test mode, only the third test sub-circuit (e.g. used to launch data so that scan cells in uncore can capture) and the second test sub-circuit (e.g. used to capture data that is launched by scan cells from uncore) may be used and the first test sub-circuit is shut off, for example.

In CPU TAP based debug mode, all these 3 sub-circuits may be accessed by CPU TAP.

For example, the flip flops of the inbound BILBO and/or the flip flops of the outbound Bil-
5 bo may be connectable to a scan chain of the SoC uncore and the flip flops of the internal
BILBO may be connectable to another or the same scan chain of the SoC uncore or a scan
chain of the CPU. These scan chains may be accessible by the SoC scan test and may be
operated with the SoC clock frequency and/or may be accessible by the CPU TAP and may
be operated with the CPU TAP clock frequency, for example.

10

In sealing mode, values may be shifted in through CPU TAP and then stop there. Then the
SoC ATPG may be started.

For example, all the modes above may be mutual exclusive.

15

More details and aspects of the system-on-chip device 200 are mentioned in connection with
the proposed concept or one or more examples described above (e.g. Fig. 1) or below. The
system-on-chip device 200 may comprise one or more additional optional features corre-
sponding to one or more aspects of the proposed concept or one or more examples described
20 above (e.g. Fig. 1) or below (e.g. Fig. 3-13).

Fig. 3 shows a possible implementation of a build-in logic block observer circuit 300 (BIL-
BO circuit). The BILBO circuit 300 comprises a control logic circuit with two control input
signals B1 and B2. The control input signals B1 and B2 are used to set the BILBO circuit
25 300 to different test modes according to table 390.

Setting "B1 = 0" and "B2 = 0", the BILBO circuit 300 is configured to operate as serial-
input serial-output shift register in a serial scan shift mode. A BILBO multiplexer 320 is set
to transfer a serial input signal "SI" into a set of BILBO data flip-flops 330. The set of BIL-
30 BO data flip-flops 330 comprises a first BILBO data flip-flop 332, a second BILBO data
flip-flop 334, a second to last BILBO data flip-flop 336, and a last BILBO data flip-flop
338, which are loaded with a frequency of an applied BILBO clock signal "Clock" and are
connected in series by a configuration of a BILBO control logic circuit 310 (the number of
BILBO data flip-flops depends on an application of the BILBO circuit 300, e.g., the number

of BILBO data flip-flops may be equal to a width of a C2U payload sub channel circuit or of a U2C payload sub channel circuit, the BILBO circuit 300 is connected to). The corresponding output signal is provided as a serial output signal "SO" by the last BILBO data flip-flop 338.

5

The number of data flip-flops, which are comprised by the set of BILBO data flip-flops 330, may be variable and be adapted to the width of a channel circuit of the input/output interface circuit, to which the BILBO circuit 300 is connected in the system-on-chip device 200.

- 10 Setting "B1 = 1" and "B2 = 0", the BILBO circuit 300 is configured to operate as parallel-input parallel-output shift register with a depth of one bit in a data-flip flop mode (D-FF mode). The parallel data input signals „D1“, „D2“, „Dn-1“, and „Dn“, are transferred, in other words, captured, into the set of BILBO data flip-flops 330 with the frequency of the clock signal Clock, and are provided as corresponding parallel data output signals Q1, Q2,
15 Qn-1, and Qn.

- Additionally, after initializing values in the set of BILBO data flip-flops 330, for example either in the serial scan shift mode or in the data-flip flop mode, the BILBO circuit 300 may be configured as a linear feedback shift register (LFSR) circuit, by setting "B1 = 0" and
20 "B2 = 0" in an LFSR-mode. In this mode, the BILBO data flip-flops, comprised by the set of BILBO data flip-flops 330, are connected in series. The BILBO multiplexer 320 is set such that a feedback signal, provided by a feedback circuit 340, which combines the parallel data output signals Q1, Q2, Qn-1, and Qn, is applied to the input of the first BILBO data flip-flop 332 and further shifted into the serially connected BILBO data flip-flops with the
25 clock signal "Clock". In this way, a pseudo-random test pattern is provided by the parallel data output signals Q1, Q2, Qn-1, and Qn.

- Another mode of the BILBO circuit 300 is a multiple-input signature register (MISR) mode, which is provided by setting "B1 = 1" and "B2 = 1". The BILBO circuit 300 is thus configured as a MISR circuit. After having reset the values stored in the set of BILBO data flip
30 flops 330, the parallel data input signal "D1" is combined with the feedback signal, provided by the feedback circuit 340 through the BILBO multiplexer 320, and the parallel data input signals "D2", "Dn-1", and "Dn", are combined with the parallel data output signal of the preceding BILBO data flip-flop, Q1, Q2, and Qn-1, respectively. In this way, the values

stored in the set of BILBO data flip-flops 330 depend on preceding values, such that a sequence of parallel data input signals "D1", "D2", "Dn-1", and "Dn", is compacted into a signature that is stored inside the set of BILBO data flip-flops 330. After having received a signature, the signature can be provided by the parallel data output signals "Q1", "Q2", "Qn-1", and "Qn", and/or the serial output signal "SO" in the serial scan shift mode.

In this example of a possible implementation, a BILBO Segment infrastructure is built on the BILBO circuitry. BILBO stands for Build-In Logic Block Observer. It is, for example, a circuit structure that may be configured into 4 different modes, which are LFSR, MISR, D-FF and Serial scan chain. A BILBO circuitry diagram is shown in Fig. 3.

In scan shift mode, the data may shift from a serial input (SI) through all the data flip-flops to a serial output SO. In LFSR mode, a BILBO may generate a pseudo-random data stream and may output from Q pins in parallel. In D-FF mode, data may be captured from D pins into corresponding data flip-flops. In MISR mode, the parallel inputs from D pins may continuously be compressed into a signature.

Fig. 4 shows a schematic illustration 400 of an input/output interface circuit (e.g. CPU and SOC interface) comprising three U2C channel circuits (which may be the U2C request channel circuit 250, the U2C response channel circuit 252, and the U2C data channel circuit 254) and three C2U channel circuits (which may be the C2U request channel circuit 260, the C2U response channel circuit 262, and the C2U data channel circuit 264). Each channel circuit comprises three sub channel circuits, namely a request (req) sub channel circuit, an acknowledge (ack) sub channel circuit, and a payload sub channel circuit, which allow a communication between the CPU and the uncore circuit 210 according to a handshaking protocol. This may allow the CPU and the uncore circuit 210 to operate independently from each other and with different clock frequencies.

Each sub channel circuits can comprise a memory circuit that is configured to be written and to be read at different clock frequencies, for example an asynchronous first-in first-out memory circuit or a bubble generation first-in first-out memory circuit. These memory circuits of the C2U sub channel circuits and the U2C sub channel circuits may have common buffer depth, for example a buffer depth of 4, 8 or 16 bit. The widths of the memory circuits of the request sub channel circuits and of the acknowledge sub channel circuits may be one

bit or more, wherein widths of one bit may be used to save die-area on the system-on-chip device 200. The widths of the memory circuits of the payload sub channel circuits may be wider than one bit in order to achieve a higher data rate of a communication between the uncore circuit 210 and the CPU.

5

For example, the communication between the uncore circuit 210 and the CPU is explained by an example of the C2U data channel circuit. To send a data signal from the CPU to the uncore circuit 210, the CPU first provides an input request signal to the input of the C2U data request sub channel circuit 460. This input request signal is written into the memory circuit of C2U data request sub channel circuit 460 at the clock frequency of the “mclk” clock signal and, after having passed the buffer, is read from the memory circuit by the uncore circuit 210 at the clock frequency of the “uclk” clock signal. In other words, the C2U data request sub channel circuit 460 provides an output request signal to the uncore circuit 210 to indicate that there is data pending to be transferred from the CPU to the uncore circuit 210.

15

When the uncore circuit 210 is ready to receive this data, it sends an output acknowledge signal to the input of the C2U data acknowledge sub channel circuit 462, where it is written into a memory circuit of this sub channel circuit 462 at a clock frequency of “uclk” and provided as an output acknowledge signal at a clock frequency of “mclk” to the CPU.

20

The CPU is now informed that the uncore circuit 210 is ready to receive the pending data and sends the corresponding input payload signal to an input of the C2U payload sub channel circuit 464. After being written into a corresponding memory circuit of the sub channel circuit 464 at clock frequency of “mclk”, the input payload signal is provided as an output payload signal at clock frequency of “uclk” at an output of the memory circuit of the sub channel circuit 464 and is received by the uncore circuit 210.

25

The exchange of signals across the other channel circuits, i.e., the C2U request channel circuit 260, the C2U response channel circuit 262, the U2C request channel circuit 250, the U2C response channel circuit 252, and the U2C data channel circuit 262, between the CPU and the uncore circuit 210 may work analogously to the example illustrated above.

30

As shown in Fig. 4, CPU core and SOC uncore may exchange data mainly through IDI interface. In this exemplary implementation, each channel may be composed of three sub channels. For instance, they are REQ (Request), ACK (Acknowledge) and PAYLOAD. Hand-shaking may be done by request/REQ from transmitter and acknowledgement/ACK from receiver. The effective data may then be sent through PAYLOAD sub channel. Each sub channel may for example be a FIFO (First-in First-Out) with a depth of 8. All U2C and C2U channels may have the same structure except that the widths of PAYLOAD sub channels may be different. ACK sub channel and REQ sub channel may be 1-bit wide FIFOs with depth of 8. Depending on the direction of each sub channel, FIFO may read and write with different clocks. For example, in any U2C channel, the REQ sub channel and PAYLOAD sub channel may write with SOC clock and read with CPU clock. For ACK sub channel, write may be done with CPU clock and read may be done with SOC clock.

For example, the test circuit of the system-on-chip device 200 may enable to test the communication between the CPU and the uncore circuit 210 across the channel circuits 250, 252, 254, 260, 262, and 264, of the input/output interface circuit. A test may be carried out under the conditions of a data transfer between the CPU and the uncore circuit 210 in functional operating conditions of the system-on-chip device 200, e.g., employing clock frequencies during the test that are also used in functional operating modes. The test may detect errors occurring in the communication between the CPU and the uncore circuit 210 in functional operating modes. For instance, the test may also cover an investigation of effects due to asynchronous clocks of CPU and uncore circuit 210 as well as high frequency effects.

In a possible implementation, the Inbound BILBO network may comprise three BILBO circuits. For each of these BILBO circuits, its parallel data outputs Q1 to Qn may be connected to a data input of a dedicated payload sub channel circuit of the U2C channel circuits. Consequently, the width of each of these BILBO circuits, (e.g. the number of parallel data outputs) may be equal to the width of its corresponding payload sub channel circuit.

For example, the parallel data inputs „D1“ to „Dn“ of each of these BILBO circuits may be connected to the outputs of the uncore circuit 210, that are designated for the CPU and connected to the same payload sub channel circuit as the parallel data outputs Q1 to Qn of that BILBO circuit.

Three BILBO circuits may be comprised by the Inbound BILBO network and may be connected in series through their serial outputs SO and serial inputs SI. For example the serial output SO of a BILBO circuit, which is connected to the U2C request channel circuit, may be connected to the serial input SI of a BILBO circuit, which is connected to the U2C response channel circuit; and the serial output SO of the latter BILBO circuit may be connected to the serial input SI of the remaining BILBO circuit, which is connected to the U2C data channel circuit. In this manner, the three BILBO circuits may form a serial chain, which may be connectable to a scan chain of the system-on-chip device 200.

For the control of the three BILBO circuits, comprised by the Inbound BILBO network, their control inputs B1 and B2 may be connected to the input control signal "Inbound B[2:1]" provided by the mode decoder circuit. Additionally, their clock inputs Clock may be connected to the "inbound_clk" clock signal delivered by the clock control circuit.

In a similar manner, the Outbound Bilbo network may also comprise three BILBO circuits in a possible implementation of a system-on-chip device. Here, each of these BILBO circuits corresponds to a dedicated C2U channel circuit, meaning that the parallel data inputs „D1“ to „Dn“ are connected to the data output of the corresponding payload sub channel circuit. Consequently, the width of each of these BILBO circuits may again be equal to the width of the corresponding payload sub channel circuit. The data outputs Q1 to Qn of each of these BILBO circuits may be connected via a corresponding multiplexer to the inputs of the uncore circuit, that are configured to receive the payload data from that C2U channel circuit, the BILBO circuit is connected to.

The three BILBO circuits, comprised by the Outbound BILBO network, may also be connected in series through their serial outputs SO and serial inputs SI, so that they can form a serial chain together with the three BILBO circuits, comprised by the Inbound BILBO network, that has the serial data input "SI from SOC" and the serial data output "SO to SOC".

For example, the control inputs B1 and B2 of the three BILBO circuits, comprised by the Outbound BILBO network, may be connected to the input control signal "Outbound B[2:1]" provided by the mode decoder circuit, whereas their clock inputs Clock may be connected to the clock signal "outbound_clk" provided by the clock control circuit.

The internal BILBO network may comprise a BILBO circuit, whose width may be the larger one of the sum of the widths of the three U2C payload sub channel circuits and the sum of the widths of the three C2U payload sub channel circuits. Its parallel data inputs „D1“ to „Dn“ are connected to the data outputs of the payload subchannel circuits of the U2C channel circuits. Its parallel data outputs Q1 to Qn may be connected via the multiplexer to the data inputs of the payload subchannel circuits of the C2U channel circuits.

The clock input of the BILBO circuit 300 comprised by the internal BILBO network may be connected to the clock signal “Internal_clk” provided by the clock control circuit. Its control inputs B1 and B2 may be connected to the input control signal “Internal B[2:1]” provided by the mode decoder circuit. Furthermore its serial input SI as well as its serial output SO may be connected to a scan chain of the CPU, which is accessible through the CPU TAP.

The test circuit comprising the Internal BILBO network, the Inbound BILBO network, and the Outbound BILBO network, can be configured to different test modes according to the table 500 displayed in Fig. 5. These different test modes of the test circuit may show an example of different tests enabled by the system-on-chip device shown in Fig. 2.

The control signal “TAP_BILBO_CFG[1:0]” may be equivalent to the control signal “GNB_config[1:0]” provided by the CPU TAP 270. The mode decoder circuit 284 receives the “TAP_BILBO_CFG[1:0]” control signal and translates it into corresponding input control signals “Inbound B[2:1]”, which comprises the inbound input control signals “B1” 510 and “B2” 511 for the Inbound BILBO network 220, “Outbound B[2:1]”, which comprises the outbound input control signals “B1” 520 and “B2” 521 for the Outbound BILBO network 230, and “Internal B[2:1]”, which comprises the internal input control signals “B1” 530 and “B2” 531 for the Internal BILBO network 240. Moreover, the inbound clock signal “CLK” 512 is equivalent to the clock signal “inbound_clk”, the outbound clock signal “CLK” 522 is equivalent to the clock signal “outbound_clk”, and the internal clock signal “CLK” 532 is equivalent to the clock signal “internal_clk”.

The test mode labelled as Function Mode may be used during a regular operation, (e.g. a functional operating mode of the system-on-chip device). In this mode, the clock signals of the Inbound BILBO network 220, the Outbound BILBO network 230, and the Internal

BIBLO network 240, are switched off, such that the contents of their data flip-flops as well as their parallel data output signals do not change and arbitrary control input signals may be applied to them. This may decrease the energy consumption of the test circuit 140. Additionally, the parallel data output signals of the Inbound BILBO network 220 and of the In-
5 internal BILBO network 240 may be disconnected from the input/output interface 130 by the use of the set of inbound multiplexers 221 and the multiplexer 242 of the CPU, respectively. The parallel data output signals of the Outbound BILBO network 230 may be disconnected from the uncore circuit 210 by the use of the set of outbound multiplexers 231. Thus the test
10 circuit 140 does not influence the communication and operation of the CPU and the uncore circuit 210, such that the system-on-chip device 200 can operate in a functional mode, for example.

Setting the control signal "TAP_BILBO_CFG[1:0]" equal to "2'b01" may be used for two different test modes of the test circuit. These are a SOC ATPG mode and a debug mode, for
15 example.

The SOC ATPG mode may be used to test at least one of the signal paths between the uncore circuit 210 and the Inbound BILBO network 220 and the signal paths between the Outbound BILBO network 230 and the uncore circuit 210. The SOC ATPG mode may be di-
20 vided into a scan phase and a capture phase. During the scan phase the Inbound BILBO network 220 and the Outbound BILBO network 230 may be configured into serial scan shift mode, so that a bit test pattern can be loaded into their BILBO data flip-flops and into the uncore circuit 210. The loading may be performed with the "SOC scan shift clock" provided by the clock control circuit 286. During the capture mode the bit test pattern may be applied
25 via the signals paths between the uncore circuit 210 and the Inbound BILBO network 220 to the Inbound BILBO network 220, and may be applied via the signals paths between the Outbound BILBO network 230 and the uncore circuit 210 to the uncore circuit 210 at the "SOC scan capture clock".

30 A possible hardware implementation for the control circuitry (e.g. external connections of BILBO network), which may be used during the SOC ATPG mode, is illustrated by the use of Fig. 6A (e.g. top level connection for BILBO network GBN viewed from CPU side) and Fig. 6B (e.g. top level connection for BILBO network GBN viewed from SOC side). In this exemplary implementation, the Inbound and Outbound BILBO networks 620 and the Inter-

nal BILBO network 240 are comprised by the CPU. The memory circuits of the C2U channel circuits and the memory circuits of the U2C channel circuits comprise bubble generation first-in first-out memory circuits (BGF), which are connected to the Inbound and Outbound BILBO networks 620 and to the Internal BILBO network 240.

5

The Inbound and Outbound BILBO networks 620 may either be controlled by the CPU TAP 270 or by the uncore circuit according to the setting of a clock multiplexer 602, a serial input multiplexer 604, and a enable multiplexer 606. The CPU TAP 270 can also be employed to control the Internal BILBO network 240, in this example.

10

The uncore circuit may provide serial input data to the Inbound and Outbound BILBO networks 620 via an uncore serial input signal "SI from SOC" at a clock signal "ATPG SCAN CLK from SOC", while the CPU can provide serial input data to the Inbound and Outbound BILBO networks 620 as well as to the Internal BILBO network 240 via the serial input signal "TAP TDI" at a clock signal "TCK", which may be the clock of the CPU TAP 270 or a scan chain of the CPU.

15

Furthermore, in order to shift serial data into and out of the Inbound and Outbound BILBO networks 620, the uncore circuit may provide an uncore enable signal "SE", while the CPU TAP 270 may provide a shift signal "TAP shift_dr".

20

The uncore circuit may read data serially from the Inbound and Outbound BILBO networks 620 via the serial output signal "SO to SOC". For example, the CPU TAP 270 may read data serially from the Inbound and Outbound BILBO networks 620 via the serial output signal "To TDO MUX". Moreover, the CPU TAP 270 may read data serially from the Internal BILBO network 240.

25

As shown in Fig. 6B, at the side of the uncore circuit 210, the uncore serial input signal "SI from SOC", the clock signal "ATPG SCAN CLK from SOC", and the uncore enable signal "SE" can be either provided from inputs of the system-on-chip device 200 (e.g. from SOC level pins) or from a Design-For-Excellence wrapper and cluster Design-For-Excellence unit, designated as CDU Cluster DFX unit 610, depending on the setting of the uncore serial input multiplexer 608. The serial output signal "SO to SOC" may be either transferred to the

30

CDU Cluster DFX unit 610 or to an output of the system-on-chip device 200 (e.g. to SOC level pins).

The table of Fig. 5 summarizes the clock and configuration signals used in different modes.

5 In this example, “se” is the inversion of SOC scan enable signal while “tap_shiftdr” is from CPU TAP. Furthermore, in this example, “ssclk” is SOC scan clock, which may be generated by multiplexing “SOC scan shift clock” and at-speed “SOC scan capture clock” pulses, in this example. In this example “tck” is the CPU TAP clock and “uclk” is the SOC function clock while “mclk” is the CPU function clock. Additionally, there may be peripheral sub-
10 blocks, for example a BGF BIST counter and req-ack debug segments.

As shown in Fig. 6A, both the CPU TAP and the SOC may have access to the inbound BILBO segment and the outbound segment. The CPU TAP may have access to the internal BILBO segment.

15

Connections to this infrastructure in SOC level are shown in Fig. 6B. The SOC may, for example, provide three inputs to the CPU, which are “SE”, “SCANCLK” and “SI”. An output “SO” may be returned from the CPU to the SOC. In this example, “SCANCLK” is a combined clock from “SOC scan shift clock” and “SOC scan capture clock”. Depending on
20 stuck at ATPG test or at-speed ATPG test, “scan capture clock” may contain one or more consecutive at-speed SOC function clock pulses. In this example, “SE” is a scan enable signal.

The CPU BILBO network may at least one of be connected to the SOC top level pins and be
25 connected into SOC scan system ring architecture through DFX wrapper/cluster DFX unit. During ATPG, a simple BILBO segment ATPG model of 1-bit chain may be plugged into an ATPG tool to generate test patterns to cover the paths between CPU and SOC.

The internal BILBO network may additionally be employed in a sealing test mode of the
30 test circuit. In this test mode, the Internal BILBO network may provide predefined signals to the CPU instead of the output signals from the U2C channel circuits and the C2U channel circuits. In other words, output signals from the U2C channel circuits and the C2U channel circuits, which are directed to the CPU, are replaced by the Internal BILBO network by predefined signals. For example, these predefined signals can be constant signals. Such con-

stant signals can, for instance, be serially loaded into the Internal BILBO network 240 by the CPU TAP 270 and be kept constant by switching off the clock signal “Internal_clk” employing the clock control circuit 286. Moreover, the Internal BILBO network 240 may sink output signals of the CPU bound for the uncore circuit 210 during the sealing test mode of the test circuit 140.

In this way, the communication between the CPU and the uncore circuit 210 can cease during the sealing test mode. This may allow testing the CPU independently from the uncore circuit 210. For such an independent test of the CPU, test signals may be applied to the CPU and received from the CPU by the CPU TAP 270 or by inputs and outputs of the system-on-chip device 200, respectively.

In other words, when running a CPU ATPG to target logic faults inside the CPU, a large number of Xs, i.e., a large number of undefined input signals, coming from the SOC interface may affect the CPU ATPG scan coverage. With the BILBO network system, this interface can be sealed during CPU ATPG test, in this example.

Setting the control signal “TAP_BILBO_CFG[1:0]” equal to “2'b10” results in a test mode of the test circuit, which may be used to test the communication via the U2C channel circuits and is referred to as inbound BGF BIST.

By setting inbound input control signals “B1” 510 equal to zero and “B2” 511 equal to one in this test mode, the Inbound BILBO network 220 may be configured as an LFSR to provide pseudo random test data to the three U2C channel circuits. In other words, it is configured to provide a test input signal to the input/output interface circuit in an inbound test mode of the test circuit.

The Internal BILBO network 240 may be configured as a MISR, by setting the internal input control signals “B1” 530 equal to one and “B2” 531 equal to one. Thus, the Internal BILBO network 240 may compact the pseudo random test data sent by the Inbound BILBO network 220 through the U2C channel circuits into a signature. This signature may be evaluated by the CPU (e.g. it may be compared to an anticipated signature) to state whether or not the U2C channel circuits work correctly.

In order to provide the clock frequency conditions of the functional mode of the system-on-chip device 200 during the inbound BGF BIST, the clock signal "inbound_clk" of the Inbound BILBO network 220 may be set to the uncore circuit 210 clock "uclk" and the clock signal "internal_clk" of the Internal BILBO network 240 may be set to CPU clock "mclk".

- 5 As the inbound BGF BIST may not employ the Outbound BILBO network 230, its clock signal "outbound_clk" may be switched off and the outbound input control signals "B1" 520 and "B2" 521 may be set arbitrarily.

- To test communication via the C2U channel circuits, the test circuit may be configured to an
 10 outbound test mode (e.g. outbound BGF BIST) by setting the control signal "TAP_BILBO_CFG[1:0]" equal to "2'b11". Thus, the mode decoder circuit 284 may set the internal input control signal "B1" 530 to zero and the internal input control signal "B2" 531 to one. This configures the Internal BILBO network 240 as an LFSR, so that it may provide pseudo random test data to the C2U channel circuits. In addition, the mode decoder circuit
 15 284 may set the outbound input control signal "B1" 520 to one and the outbound input control signal "B2" 521 to one, which, in this example, results in a configuration of the Outbound BILBO network 230 as a MISR to receive the pseudo random test data via the C2U channel circuits, sent by the Internal BILBO network 240.

- 20 Again, in order to provide the clock frequency conditions of the functional mode of the system-on-chip device 200 also during the outbound BGF BIST, the clock signal "outbound_clk" of the Outbound BILBO network 230 may be set to the uncore circuit 210 clock "uclk" and the clock signal "internal_clk" of the Internal BILBO network 240 may be set to CPU clock "mclk".

25

In the outbound BGF BIST, the Inbound BILBO network 220 may not be employed, so that its clock signal "inbound_clk" may be switched off and the inbound input control signals "B1" 510 and "B2" 511 may be set arbitrarily.

- 30 Referring to Fig. 7, the outbound BGF BIST 700 is explained in more detail below. Fig. 7 shows a C2U channel circuit 710, also referred to as C2U tunnel, which may be an example for the C2U request channel circuit 260, the C2U response channel circuit 262, and/or the C2U data channel circuit 264 of Fig. 2. The C2U channel circuit 710 comprises a C2U request sub channel circuit 712, a C2U acknowledge sub channel circuit 714, and a C2U pay-
 35 load sub channel circuit 716.

Further, an internal LFSR 750 (e.g., the Internal BILBO network may be configured as the internal LFSR), an outbound MISR 760 (e.g. the Outbound BILBO network may be configured as the outbound MISR), and the clock control circuit 286 to provide a clock signal
5 “MISR clk”, which may be equivalent to “uclk”, to the outbound MISR 760 and a clock signal “LFSR clk”, which may be equivalent to “mclk”, to the internal LFSR 750 are shown.

During outbound BGF BIST a first outbound BGF BIST multiplexer 732, which may be
10 comprised by the multiplexer 242 of the CPU, may be configured to transfer a BIST request signal to the outbound MISR 760. A BIST request signal, containing a logical one, indicates to the outbound MISR 760, that there is new incoming pseudo random test data provided by the internal LFSR 750.

15 The BIST request signal may be generated by a first internal AND-Gate 728, which may be comprised by the test control circuit 280. The first internal AND-Gate 728 may have four inputs, that are all at a logic value of one, in order to generate a BIST request signal indicating new incoming pseudo random test data for the outbound MISR 760.

20 A first input signal to the first internal AND-Gate 728 may be the signal “BIST_go”. “BIST_go” may be a trigger signal provided by the BGF BIST Creg 272 to start and to stop the outbound BGF BIST.

A second input signal to the first internal AND-Gate 728 may be provided by the 4-bit valid
25 rotate register 726. For example, the 4-bit valid rotate register 726 can be regarded as a 4-bit serial shift register, whose serial output is connected to its serial input, such that its contents can circle through its data flip-flops. By connecting the serial output of the 4-bit valid rotate register 726 also to the first internal AND-Gate 728, the input signal provided by the 4-bit valid rotate register 726 to the first internal AND-Gate 728 may change periodically. This
30 may allow modulating the pseudo random test data transfer speed during the outbound BGF BIST. For example, loading the 4-bit valid rotate register 726 with a value of “4'b1111” (e.g. all data-flip-flops of the 4-bit valid rotate register 726 contain logical ones), the second input signal to the first internal AND-Gate 728, may always be at logical one. Consequently, the pseudo random test data can be transferred at full speed (e.g. at the speed of functional

mode of the system-on-chip device). If, for example, the 4-bit valid rotate register 726 also contains logical zeros, for instance, when it is initialized with a value of "4'b0001", the second input signal to the first internal AND-Gate 728 will at times be at logical zero, and thus slow down the pseudo random test data transfer. This may be accomplished by a connection of the "BIST request signal" to an enable input of the internal LFSR 750, that stops the output of pseudo random test patterns, when a logical zero is applied to it.

Slowing down the transfer of the pseudo random test data may be used for debugging and error analysis. For example, if an erroneous behavior of the pseudo random test data transfer occurs at full speed, the pseudo random test data transfer may be slowed down, in order to investigate whether the erroneous behavior is due to a too high speed. This may be an indication for high frequency effects that can disturb the data transfer via the channel circuit under investigation.

A third input signal to the first internal AND-Gate 728 may be provided by a zero detector, which may comprise the internal XNOR-Gate 274 (or XOR) and which may be asserted, when the number of pseudo random test patterns, generated by the internal LFSR 750 and counted by the counter circuit 282, reaches a specified pattern count in a max counter control register 722, which may be comprised by the BGF BIST Creg 272. By a connection of the "BIST request signal" to an enable input of the counter circuit 282, only these clock cycles of the clock signal "LFSR clk", which is also connected to the counter circuit 282, may be counted, during which pseudo random test data is provided by the internal LFSR 750. In this example, the counted number of the counter circuit 282 is compared to the specified pattern count in the max counter control register 722 by the use of the internal XNOR-Gate 724 (or XOR), which provides a logical zero at its output, when the counter reaches the specified pattern count, otherwise it provides a logical one to let the transfer of pseudo random test data run. The third input signal to the first internal AND-Gate 728 may be used to stop the transfer of pseudo random test data, when the specified pattern count is reached. In another example, the counter 282 may also be configured as a countdown circuit, which may be initialized with the specified pattern count in the max counter control register 722, and which stops the transfer of pseudo random test data, when it reaches zero.

A fourth input signal to the first internal AND-Gate 728 may be provided by a credit manager circuit 736. The credit manager circuit 736 may have a maximum number of credits

- equal to the common buffer depth of the memory circuits comprised by the C2U request sub channel circuit 712, the C2U acknowledge sub channel circuit 714, and the C2U payload sub channel circuit 716. An outbound counter circuit 738, which may be connected to the C2U request sub channel circuit 712, may count the number of logical ones inside the memory circuit of the C2U request sub channel circuit 712, which may be equivalent to the number of pseudo random test patterns pending inside the memory circuit of the C2U payload sub channel circuit 716. The outbound counter circuit 738 may communicate this number to the credit manager circuit 736, such that the credit manager circuit 736 may decrease its credits by this number. Consequently, when, for example, the buffer of the memory circuit of the C2U payload sub channel circuit 716 is full and hence data cannot be anymore transferred into it, the credits of the credit manager circuit 736 reach zero. The credit manager circuit 736 may then provide a logical zero to the first internal AND-Gate 728 to stop the transfer of pseudo random test patterns into the C2U payload sub channel circuit 716.
- When, for instance, the outbound counter circuit 738 reads a logical one from the C2U request sub channel circuit 712, it may also provide a logical one at an output connected to an enable input of the outbound MISR 760. The outbound MISR 760 may read a pseudo random test pattern from the memory circuit of the C2U payload sub channel circuit 716.
- In accordance, the outbound counter circuit 738 may inform the credit manager circuit 736 via a signal “irdy”, that a pseudo random test pattern has been read by the outbound MISR 760 and thus an empty slot in the memory circuit of the C2U payload sub channel circuit 716 and an empty slot in the memory circuit of the C2U request sub channel circuit 712 are available. In consequence, the credit manager circuit 736 may increase its credits by one to enable a transfer of a new pseudo random test pattern.

During the outbound BGF BIST the “irdy” signal is transferred via a second outbound BGF BIST multiplexer 742, an outbound AND-Gate 740 and the U2C acknowledge sub channel circuit 714 to the credit manager circuit 736. As the data is read by the outbound MISR 760, which may at all clock cycles of its applied clock signal “MISR clk” be ready to read during outbound BGF BIST, and not by the uncore circuit 210, the “irdy” signal may be directly fed back to the C2U acknowledge subcircuit 714 and may not have to be subjected to a logical AND-operation at the outbound AND-Gate 740 with a “functional trdy” signal of the uncore circuit 210.

During functional mode of the system-on-chip device 200, the “irdy” signal may indicate to the uncore circuit 210, that data can be read from the C2U payload sub channel circuit 716. The uncore circuit 210 may answer through the “functional trdy” signal, that is at logical one, when it is ready to receive this data via the “payload data to SSA” signal. The “functional trdy” signal may thus be transferred through the second outbound BGF BIST multiplexer 742 to the outbound AND-Gate 740. Now both “irdy” and “functional trdy” may be at logical one at the inputs of the outbound AND-Gate 740, such that it outputs a logical one to the C2U acknowledge sub channel circuit 714. In turn, the credits of the credit manager circuit 736 may be increased by one. The credit manager circuit 736 may then output a logical one, which is applied to a second internal AND-Gate 746. Another input signal to the second internal AND-Gate 746, may be a signal “bist_mode_inv”. During functional mode “bist_mode_inv” may be at logical one, while during outbound BGF BIST it may be at logical zero. Hence, during functional mode, the second internal AND-Gate 746 may output a logical one, if the credit manager circuit 736 still has credits, which indicates to the CPU, in other words, to the eb fubs of the CPU, that data can be written into the C2U channel circuit 710. In turn, when the CPU has data to transfer to uncore circuit 210, it may send a “functional valid” at logical one through the first outbound BGF BIST multiplexer 732 to the C2U request sub channel circuit 712, and functional payload data to the C2U payload sub channel circuit 716 through a third outbound BGF BIST multiplexer 734.

According to another aspect, the BILBO infrastructure may be configured for a BGF BIST. A BGF BIST may be divided into two passes, which comprise an inbound BGF BIST and an outbound BGF BIST. The inbound BGF BIST covers, for example, the three U2C channels, while the outbound BGF BIST covers, for example, the three C2U channels. During the inbound BIST, the inbound BILBO may be configured as an LFSR, which provides random test data to the U2C channels, while the internal BILBO may be configured as a MISR to compact a payload data slice coming out from U2C channels. Analogously, during the outbound BIST, the internal BILBO may be used as an LFSR, while the outbound BILBO may be used as a MISR. In both inbound and outbound BIST application, the hand-shaking process may still be handled via req and ack protocol, which, in this example, is same as the functional mode. Thus, both the BGF arrays as well as the BGF control may be covered through BGF BIST. The outbound BGF BIST is now illustrated by an example of a C2U channel. For U2C channels, the outbound BIST process may be similar.

From the transmitter side, a BIST REQ signal may be multiplexed with a functional valid through a multiplexer, before it is sent to a C2U req sub-tunnel to inform the receiver/MISR of the new incoming data. In BGF BIST mode, this REQ signal may be generated by AND-
5 ing four signals:

Firstly, in this example, "BIST_go" is a trigger signal from Creg, which starts the BIST process.

10 Secondly, a 4-bit valid rotate register may be used to modulate the data transfer speed during BIST. By default, it may be loaded with 4'b1111, which allows the BGF to go full-speed. During debug, if needed, a smaller value, such as 4'b0001, may be loaded to slow down the effective data transferring.

15 Thirdly, the third signal may be from a zero detector, which, in this example, is asserted, when the number of generated random patterns by LFSR reaches a specified pattern count in a max counter control register (creg). It may be used to stop the transmission, when the specified pattern count is reached.

20 Fourthly, the 4th signal may be from a credit manager of the ack sub-tunnel. In this exemplary implementation, when the credit manager runs out of credit and FIFO is full, then it zeros out the REQ signal to stop new data transferring temporarily, until FIFO has new slot after some data in FIFO is consumed by the MISR.

25 This REQ signal may also be sent to the LFSR as an enable signal to generate the next data slice. From the receiver side, an "irdy" signal, coming out of C2U req sub-channel, may directly be assigned back to the ack sub-channel during BIST mode, without waiting for a "trdy" signal from SOC side, because the MISR, in this example, is always ready in BGF BIST, whenever new data comes out.

30

In the sequence above, when TAP instructions are used to shift in and shift out data from BILBO segments, a capture operation shall not be triggered by "TAP capture_dr". However, during the BGF debug and dump mode, "capture_dr" will be required. So a qualifier from

Creg is not to turn on or gate off the “capture_dr” of the TAP instructions depending on the usage.

Moreover, the “en” of the LSFR and the MISR shown in Figure 7 may be implemented by
5 using clock gating/enabling techniques.

Fig. 8 shows a possible implementation of a U2C sub channel circuit 800 together with the boundary of the clock domain crossing 840 and the boundary of the voltage domain crossing 850 between a CPU and an uncore circuit. In this example the U2C sub channel circuit
10 800 comprises a bubble generation first-in first-out memory circuit.

An uncore bubble generator circuit 824 is connected to a write logic circuit 822 and a first-in first-out memory circuit 812 and configured to indicate to the write logic circuit 822 and the first-in first-out memory circuit 812, when data can be written into the first-in first-out
15 memory circuit 812 through the write logic circuit 822 by the uncore circuit.

The uncore bubble generator circuit 824, the write logic circuit 822, and the first-in first-out memory circuit 812 are operated by the clock signal “uclk” of the clock domain of the uncore circuit and in a voltage domain “Vnn”, which is the voltage domain of the uncore circuit.
20

On the CPU side, a CPU bubble generator circuit 834 is connected to a read pointer circuit 836 and a CPU AND-Gate 832. The CPU bubble generator 834 is configured to indicate to the read pointer circuit and the CPU AND-Gate 832, when data can be read from the first-in first-out memory circuit 812. The read pointer circuit 836 is connected to a read multiplexer 814, which is connected to parallel data outputs of the first-in first-out memory circuit 812. The read pointer circuit 836 may thus be configured to select via the read multiplexer 814 a certain data output of the first-in first-out memory circuit 812, from which data can be read by the CPU.
25

30 The CPU bubble generator 834 circuit, the read pointer circuit 836, the CPU AND-Gate 832, and the read multiplexer 814, are operated by the clock signal “mclk” of the clock domain of the CPU and in a voltage domain “Vcc”, which is the voltage domain of the CPU.

In other words, the boundary between the clock domains 840 of “uclk” and “mclk”, and the boundary between the voltage domains 850 of “Vnn” and “Vcc” are both placed between the data outputs of the first-in first-out memory circuit 812 and the read multiplexer 814. The transition from “Vnn” to “Vcc” may be implemented by a level shifter circuit (not shown in Fig. 8).

As a higher clock frequency may need a higher voltage, and a lower clock frequency may also operate at a lower voltage, letting the boundary between the clock domains 840 coincide with the boundary between the voltage domains 850, may allow altering the clock frequency of “uclk” coupled to the voltage “Vnn” and the clock frequency of “mclk” coupled to the voltage “Vcc” independently of each other. Thus, depending on where fast data processing performance is needed or on where energy can be saved in an idle state, the clock frequency in either the CPU or the uncore circuit may be changed independently.

Another test mode of the test circuit is a debug mode, which may be employed to read out data pending inside the memory circuits of at least one of the U2C channel circuits and the C2U channel circuits.

Referring to the table of Fig. 5, the test circuit may be configured into this debug mode by setting the test circuit control signal “TAP_BILBO_CFG[1:0]” equal to “2'b01”.

An example of a debug mode is explained with respect to Fig. 9 (e.g. BGF dump through BILBO segments), which illustrates a block diagram of a possible hardware implementation to support the debug mode. As already explained in the context of Fig. 2 and Fig. 3, the Inbound BILBO network 220 and the Outbound BILBO network 230 may each comprise three BILBO circuits 300, wherein each BILBO circuit 300 is connected to a dedicated payload sub channel circuit of the U2C channel circuits and to a dedicated payload sub channel circuit of the C2U channel circuits, respectively.

This is illustrated in Fig. 9. The Inbound BILBO circuit 220 comprises an Inbound Data BILBO circuit 924, labelled as “bilboz for U2C_Data”, whose parallel data outputs “Q1” to “Qn” are connected to the data input of the payload sub channel circuit of the U2C Data channel circuit 254, an Inbound Response BILBO circuit 922, labelled as “bilboz for U2C_Rsp”, whose parallel data outputs “Q1” to “Qn” are connected to the data input of the

payload sub channel circuit of the U2C Response channel circuit 252, and an Inbound Request BILBO circuit 920, labelled as “bilboz for U2C_Req”, whose parallel data outputs “Q1” to “Qn” are connected to the data input of the payload sub channel circuit of the U2C Request channel circuit 250.

5

Analogously, the Outbound BILBO circuit 230 comprises an Outbound Request BILBO circuit 930, labelled as “bilboz for C2U_Req”, whose parallel data inputs “D1” to “Dn” are connected to the data output of the payload sub channel circuit of the C2U Request channel circuit 260, an Outbound Data BILBO circuit 934, labelled as “bilboz for C2U_Data”,
10 whose parallel data inputs “D1” to “Dn” are connected to the data output of the payload sub channel circuit of the C2U Data channel circuit 264, and an Outbound Response BILBO circuit 932, designated as “bilboz for C2U_Rsp”, whose parallel data inputs “D1” to “Dn” are connected to the data output of the payload sub channel circuit of the C2U Response channel circuit 262.

15

Moreover, the Inbound Data BILBO circuit 924, the Inbound Response BILBO circuit 922, the Inbound Request BILBO circuit 920, the Outbound Request BILBO circuit 930, the Outbound Data BILBO circuit 934, and the Outbound Response BILBO circuit 932 may be connected in series through their serial data outputs “SO” and their serial data inputs “SI” in
20 this order. The first BILBO circuit in this serial connection may thus be the Inbound Data BILBO circuit 924, whose serial input “SI” may be connected to a scan chain of the uncore circuit 210 or to a scan chain of the CPU, in particular to the CPU TAP 270. The last BILBO circuit of the serial connection is the Outbound Response BILBO circuit 932, whose serial output “SO” may be connected to the scan chain of the uncore circuit 210 via a SOC
25 AND-Gate 902 and to the scan chain of the CPU, in particular to the test data output signal “TDO” of the CPU TAP 270, via an external peripheral BILBO circuit 912, designated as “peripheral bilboz for debug”, and an external multiplexer 904.

As, in this possible implementation, the Outbound Request BILBO circuit 930, the Out-
30 bound Data BILBO circuit 934, and the Outbound Response BILBO circuit 932 are connected to the payload sub channel circuits of the U2C channel circuits, they can neither read the output request signals nor the output acknowledge signals from the C2U channel circuits. Therefore, a function of the external peripheral BILBO circuit 912 may be to read the output request signals and the output acknowledge signals from the C2U channel circuits

through an input designated as “3-bit U2C acks + 3-bit C2U reqs”. By the use of the external multiplexer 904, it may be decided whether the output request signals and the output acknowledge signals from the C2U channel circuits are sent to the scan chain of the CPU or be omitted. For instance, in a possible implementation of the test modes of the test circuit, the output request signals and the output acknowledge signals may be transferred to the scan chain of the CPU during the debug mode and may be omitted (e.g. not transferred to the scan chain of the CPU) during the SOC ATPG mode.

A function of the SOC AND-Gate 902 may be to forward the serial output signal “SO” of the Outbound Response BILBO circuit 932 during the SOC ATPG mode and to block it in the debug mode. This may be controlled with a second input signal to the SOC AND-Gate 902.

During the debug mode, the Outbound Request BILBO circuit 930, the Outbound Data BILBO circuit 934, and the Outbound Response BILBO circuit 932 may first read (e.g. capture the data pending inside the payload sub channel circuits of the C2U channel circuits), while they are configured in D-FF mode. The external peripheral BILBO circuit 912, also configured in D-FF mode, may capture the output request signals and the output acknowledge signals of the C2U request sub channel circuits and of the C2U acknowledge sub channel circuits, respectively. Next, the mode of the Outbound Request BILBO circuit 930, the Outbound Data BILBO circuit 934, the Outbound Response BILBO circuit 932, and the external peripheral BILBO circuit 912 can be changed to serial scan shift mode in order to transfer the captured data to the scan chain of the CPU or to the scan chain of the uncore circuit 210.

25

Analogously, the internal BILBO network 240 may be employed to capture the data pending inside the payload sub channel circuits of the U2C channel circuits in the debug mode of the test circuit. Additionally, an internal peripheral BILBO circuit 914 can be used to capture the output request signals and the output acknowledge signals of the of the U2C request sub channel circuits and of the U2C acknowledge sub channel circuits, respectively, at an input of the internal peripheral BILBO circuit 914, designated as “3-bit C2U acks + 3-bit U2C reqs”. The captured data can then be transferred to the scan chain of the CPU, i.e., to the CPU TAP 270, via an internal multiplexer 906 by configuring the BILBO circuits comprised by the internal BILBO network 240 (not shown in Fig. 9) and the internal peripheral

30

BILBO circuit 914 into serial scan shift mode. Moreover, the internal peripheral BILBO circuit 914 may be bypassed by use of the internal multiplexer 906.

For example, the BILBO network of this example is used to dump BGFs for debug. In comparison, other CPU cores may use LDAT (Local Data Access Test Port) to support BGF array dumping. LDAT may be a CPU array DFT (Design for Testability) block, which is a gateway between local arrays and the CPU CRAB bus (Control Register Access Bus). LDAT works only on CPU clock. Thus, in such system-on-chip devices, BGF arrays are treated as regular arrays by forcing both BGF read and write clocks to be the CPU clock during LDAT mode.

As a faster clock domain may require a higher voltage and a slower clock domain only needs a lower voltage, but can also use a higher voltage, in such system-on-chip devices the voltage of the CPU core is higher than the voltage of the SOC, and the CPU clock is faster than the SOC clock. In such system-on-chip devices, both the CPU clock and the SOC clock may be put into the higher voltage domain of the CPU core. In other words, the BGFs are operated in the voltage domain of the CPU core and the crossing of the voltage domains from the SOC voltage domain to the CPU voltage domain is on the SOC side and in front of the BGFs, when viewed from the SOC. This may work well in both functional mode and LDAT mode of such system-on-chip devices: In functional mode, both the CPU clock domain and the SOC clock domain may use the higher voltage domain of the CPU. In LDAT mode, the write clock and the read clock are forced to the CPU clock, which may work well in the voltage domain of the CPU.

To further reduce power in an example, the CPU clock "Mclk" may be slower than the clock of the SOC "Uclk" and the voltage "Vcc" of the CPU may be lower than the voltage "Vnn" of the SOC, when the CPU core is in an idle state, for example. However, then the scheme of the previously mentioned system-on-chip devices does not work in the functional mode of the proposed example, because a faster clock of the SOC "Uclk" does not work in a lower voltage domain "Vcc" of the CPU, for example. Thus, a level shifter may be put in the middle of the BGFs to make both clock domain crossing and voltage domain crossing in the same border line, which is shown in Fig. 8, for example. However, that may break the LDAT, in other words, the LDAT is not applicable anymore, when the CPU clock "Mclk"

may be configured back to be faster than the SOC clock “Uclk”, because the CPU clock “Mclk” is not used in the lower voltage domain “Vnn” of the SOC, for example.

Hence, in an example according to a proposed aspect, the BGF LDAT may be replaced with the BILBO network to support array dump. As shown in Fig. 9, the BGF data may be captured into the BILBO network and may be shifted out via TAP instructions.

For example, after specifying a target dumping address via a Creg, TAP may be used to capture the payload of the three U2C channels into the internal BILBO segment and the payload of the three C2U channels into the outbound BILBO segment. Concerning the ACK and REQ signals, in this example they may be internal signals of the BGFs and may not be exposed to the internal BILBO segment and to the external BILBO segments (e.g. the Inbound BILBO network and the Outbound BILBO network). Thus, in this example, two peripheral chains at the end of the external BILBO segment and the internal BILBO segment are appended to capture the ACK and REQ signals of those six BGF channels, in other words, of the three U2C channels and the three C2U channels. The two peripheral chains may be bypassed during the functional mode and the BIST modes.

Fig. 10 shows a block diagram of a system-on-chip device according to an example. The system-on-chip device 1000 comprises a first circuit 110, a second circuit 120 and a test circuit 140 connected to an input/output interface circuit 130. The input/output interface circuit 130 is configured to receive input signals from the first circuit 110 and provide corresponding output signals to the second circuit 120. Further, the input/output interface circuit 130 is configured to receive input signals from the second circuit 120 and provide corresponding output signals to the first circuit 110. The test circuit 140 is configured to provide a test input signal to the input/output interface circuit 130 and configured to receive a corresponding test output signal from the input/output interface circuit 130.

By implementing a test circuit to provide test input signals to an I/O interface between circuits and receive corresponding test output signals from the I/O interface, the I/O interface can be tested under real operating conditions. In this way, the reliability of testing highly integrated circuits may be increased and/or the test efforts and/or test time for highly integrated circuits may be reduced.

Optionally, the test circuit 140 may be configured to provide the test input signal containing test input data at a write frequency and may be configured to read the test output signal at a read frequency. The write frequency may be higher than the read frequency in a first test mode. For example, the write frequency may be equal to a clock domain frequency of the first circuit 110 and the read frequency may be equal to a clock domain frequency of the second circuit 120 or vice versa. Further, the write frequency may be lower than the read frequency in a second test mode.

More details and aspects of the system-on-chip device 1000 are mentioned in connection with the proposed concept or one or more examples described above (e.g. Fig. 1) or below. The system-on-chip device 1000 may comprise one or more additional optional features corresponding to one or more aspects of the proposed concept or one or more examples described above (e.g. Fig. 1-9) or below (e.g. Fig. 11-13).

Fig. 11 shows a block diagram of a system-on-chip device 1100 according to an example. The system-on-chip device 1100 comprises a first core module 1102 (e.g. comprising two core circuits and a cache circuit) connected to a system agent circuit 1112 of an uncore circuit through a first input/output interface circuit 1106 and a second core module 1104 (e.g. comprising two core circuits and a cache circuit) connected to the system agent circuit 1112 of the uncore circuit through a second input/output interface circuit 1108. A first test circuit 1110 is connected to the first input/output interface circuit 1106 and a second test circuit 1118 is connected to the second input/output interface circuit 1108. Further, the uncore circuit comprises an image signal processing circuit 1120, an high definition HD graphics circuit 1122, a video decode engine circuit 1124, a display controller circuit 1126, a first memory circuit 1116 (e.g. DDR 3 memory channel 0) and/or a second memory circuit 1114 (e.g. DDR 3 memory channel 1) connected to the system agent circuit 1112 in a north cluster. Further, the uncore circuit comprises a low speed peripheral fabric circuit 1128 of a south cluster connected to the system agent circuit 1112. Further, the uncore circuit comprises a general purpose input/output GPIO circuit 1130, a storage circuit 1134, an audio engine circuit, an universal serial bus high speed inter chip USB HSIC/ USB2/USB3 circuit 1138 and/or other circuits 1132 connected to the low speed peripheral fabric circuit 1128.

The first test circuit 1110 and/or the second test circuit 1118 may be implemented according to the proposed concept or one or more examples described above or below.

More details and aspects of the system-on-chip device 1100 are mentioned in connection with the proposed concept or one or more examples described above (e.g. Fig. 1) or below. The system-on-chip device 1100 may comprise one or more additional optional features
5 corresponding to one or more aspects of the proposed concept or one or more examples described above (e.g. Fig. 1-10) or below (e.g. Fig. 12-13).

Fig. 12 shows a flow chart of a method for testing a system-on-chip device according to an example. The method 1200 comprises operating 1210 a first circuit in a first clock domain
10 and operating 1220 a second circuit in a second clock domain. A clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device. Further, the method 1200 comprises at least one of providing 1230 a test input signal to the input/output interface circuit and receiving 1240 a
15 test output signal from an input/output interface circuit in a test mode of the system-on-chip device. The input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding output signals to the second circuit and to receive input signals from the second circuit and provide corresponding output signals to the first circuit.

20 More details and aspects of the method 1200 are mentioned in connection with the proposed concept or one or more examples described above (e.g. Fig. 1) or below. The method 1200 may comprise one or more additional optional features corresponding to one or more aspects of the proposed concept or one or more examples described above (e.g. Fig. 1-11) or below (e.g. Fig. 13).

25 Fig. 13 shows a flow chart of another method for a testing system-on-chip device according to an example. The method 1300 comprises providing 1310 a test input signal to the input/output interface circuit. The input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding
30 output signals to the second circuit and to receive input signals from the second circuit and provide corresponding output signals to the first circuit. Additionally, the method 1300 comprises receiving 1320 a corresponding test output signal from the input/output interface circuit.

More details and aspects of the method 1300 are mentioned in connection with the proposed concept or one or more examples described above (e.g. Fig. 10) or below. The method 1300 may comprise one or more additional optional features corresponding to one or more aspects of the proposed concept or one or more examples described above (e.g. Fig. 1-12) or
5 below.

Some examples relate to a Unified Test and Debug Architecture for the interface between Atom CPU and SoC. For example, a Bubble-Generation-FIFO/BGF test, a CPU interface test, a debug test, a Design-For-Debug/DFD, a Design-For-Testability/DFT and/or an Intra-
10 Die-Interconnection/IDI test may be enabled based on the proposed concept.

It is a challenging task to test and debug the interface between CPU and SOC. For example, functional tests may be used to cover this interface, which is quite costly from test development perspective. It may affect the test cost and time to market. Structure-based tests such
15 as ATPG scan test and LDAT test may be used to cover the logic and BGF array respectively. But neither of them covers this interface well. For ATPG scan, scan testing runs in CPU core and SOC separately, for example, because they may have independent scan systems. CPU core and SOC may use different types of scan cells and scan architectures. That may leave the interface between CPU and SOC uncovered by ATPG test. For the BGF arrays
20 inside CPU core, they may be treated as regular arrays and may be tested with LDAT (Local Data Access Test Port) after forcing the read and write clock to be same as CPU clock, that test condition may be very different from the requirement in functional mode, which may use different clocks for read and write. Depending on the data transfer direction, in functional mode the BGFs may either write data with CPU clock and read data with SOC clock
25 or vice versa. Therefore, LDAT method that use CPU clock for both read and write may cause big overkill for BGF array testing. Moreover LDAT does not cover the BGF write and read pointer generation and control logic, for example, which may be a key element to ensure the deterministic data transferring between CPU clock domain and SOC clock domain. Alternatively, BGF BIST may be used. It may test these special FIFO arrays inside CPU.
30 But it does not test the paths between SOC and CPU, for example. Moreover, the BGF BIST does not support array dumping and it may be necessary to rely on LDAT to dump out the array content from specified entry of BGF for debug purpose. This may be difficult for CPUs supporting a wide range of CPU/SOC frequency and voltage ratios, e.g. in one condition CPU may use slower clock and lower voltage than those of SOC uncore block while in

the other condition CPU clock may be faster than SOC and CPU voltage may be higher than SOC. It may be desirable to have a single infrastructure that supports (i) BGF BIST, (ii) at-speed scan testing for the paths between SOC and CPU, (iii) BGF dump capability so that LDAT may be removed from BGF and (iv) sealing the CPU boundary during in CPU internal ATPG, for example.

According to an aspect, Goldmont BILBO Network infrastructure may be built on the BILBO circuitry, which can be configured into 4 different modes including LFSR/random pattern generator, MISR /response compactor, D-FF and Serial shift chain. For example, 3 BILBO segments may be deployed in this infrastructure, which are inbound BILBO segment, outbound BILBO segment and internal BILBO segment. (i) In BGF BIST for inbound FIFO, inbound BILBO may serve as LFSR while internal BILBO may serve as MISR. (ii) In BGF BIST for outbound FIFO, internal BILBO may serve as LFSR and outbound BILBO may serve as MISR. (iii) In testing the paths between CPU and SOC, inbound BILBO and outbound BILBO may be configured into 1-bit single chain and hooked into SOC scan system. (iv) In debugging/dumping the FIFO content, internal BILBO may be used to capture content from 3 inbound FIFOs while outbound BILBO may be used to capture content from 3 outbound FIFOs. The capture content in BILBOs may be shift out through CPU TAP in this debug mode, for example.

For example, the data exchange between CPU (central processing unit) core and SOC (system on chip) may be mainly done through an IDI (Intra Die Interconnect) interface. Multiple BGFs (Bubble Generating First-In-First-Out FIFO) may be deployed inside the CPU to support IDI. For example, a unified DFX (design for X) architecture (e.g. called GLM BILBO Network) may be proposed, which can be configured into multiple modes to support testing and debugging of this interface. With this unified architecture, the BGFs may be tested inside CPU with functional BGF BIST (Built-In Self Test). This infrastructure may be also accessible through SOC/NON-CPU scan system in the second mode such that the paths between SOC and CPU core may be testable at-speed with SOC ATPG (automatic test pattern generation) test. Moreover, this infrastructure may be also accessible through CPU TAP (test access port) in the third mode such that the content of BGF FIFO arrays can be dumped out for debug. Further, this infrastructure may be used to seal the CPU boundary during the CPU internal scan testing with the fourth mode. So this single infrastructure may satisfy all of the four different test and debug needs for CPU/SOC IDI interface, for example.

Some applications of the proposed concept relate to High Volume Architecture and/or micro architectures. The proposed concept may be embodied in computer system architecture features & interfaces made in high volumes and/or may encompass IA (integrated architectures), devices (e.g., transistors) and associated mfg. (manufacturing) processes.

In the following, some examples are described. Example 1 is a system-on-chip device comprising a first circuit configured to operate in a first clock domain, a second circuit configured to operate in a second clock domain, wherein a clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device, an input/output interface circuit configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit and configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit, and a test circuit connected to the input/output interface circuit and configured to at least one of providing a test input signal to the input/output interface circuit and receiving a test output signal from the input/output interface circuit in a test mode of the system-on-chip device.

In example 2, the subject matter of example 1 can optionally include the clock frequency of the second clock domain being higher than the clock frequency of the first clock domain in a second operating state of the system-on-chip device.

In example 3, the subject matter of example 2 or 3 can optionally include the test circuit comprising a first test sub-circuit connected to the input/output interface circuit.

In example 4, the subject matter example 3 can optionally include the first test sub-circuit being configured to provide a test input signal to the input/output interface circuit in an input test mode of the first test sub-circuit.

In example 5, the subject matter of one of example 4 can optionally include the test input signal being a random or pseudo-random bit sequence.

In example 6, the subject matter of one of the examples 3-5 can optionally include the first test sub-circuit being configured to receive a test output signal from the input/output inter-

face circuit and provide an output test signal to an output of the system-on-chip device or a scan chain of the system-on-chip device in an output test mode of the first test sub-circuit.

5 In example 7, the subject matter of example 6 can optionally include the first test sub-circuit configured as a multiple-input signature register circuit in the output test mode or comprising a multiple-input signature register circuit, wherein the multiple-input signature register circuit is configured to generate the output test signal of the first test sub-circuit indicating a signature of the test output signal of the input/output interface circuit in the output test mode.

10

In example 8, the subject matter of one of the examples 3-7 can optionally include the first test sub-circuit comprising a shift register or the first test sub-circuit is configured as a shift register in a scan test mode of the first test sub-circuit, wherein the first test sub-circuit is configured to connect the shift register to a scan chain of the system-on-chip device in the scan test mode.

15

In example 9, the subject matter of example 8 can optionally include the first test sub-circuit being configured to capture output data of at least one of the first circuit, the second circuit and the input/output interface circuit by the shift register in a capture test mode of the first test sub-circuit.

20

In example 10, the subject matter of one of the examples 1-9 can optionally include the test circuit being configured to provide predefined signals to at least one of the first circuit and the second circuit instead of the output signals of the input/output interface circuit in a sealing test mode of the test circuit in a test operation of at least one of the first circuit and the second circuit.

25

In example 11, the subject matter of example 10 can optionally include the predefined signals being constant signals in the sealing test mode of the test circuit.

30

In example 12, the subject matter of one of the examples 3-11 can optionally include the first test sub-circuit comprising a build-in logic block observer circuit.

In example 13, the subject matter of one of the examples 3-12 can optionally include the test circuit comprising a second test sub-circuit connected to the input/output interface circuit.

5 In example 14, the subject matter of example 13 can optionally include the second test sub-circuit being configured to provide a test input signal to the input/output interface circuit in an inbound test mode of the test circuit, wherein the first test sub-circuit is configured to receive a test output signal from the input/output interface circuit based on the test input signal provided by the second test sub-circuit in the inbound test mode of the test circuit.

10 In example 15, the subject matter of example 13 or 14 can optionally include the test circuit comprising a third test sub-circuit connected to the input/output interface circuit.

15 In example 16, the subject matter of example 15 can optionally include the first test sub-circuit being configured to provide a test input signal to the input/output interface circuit in an outbound test mode of the test circuit, wherein the third test sub-circuit is configured to receive a test output signal from the input/output interface circuit based on the test input signal provided by the first test sub-circuit in the outbound test mode of the test circuit.

20 In example 17, the subject matter of one of the examples 15-16 can optionally include the second test sub-circuit comprising a shift register or the second test sub-circuit is configured as a shift register in a scan test mode of the test circuit, wherein the third test sub-circuit comprises a shift register or the third test sub-circuit is configured as a shift register in the scan test mode of the test circuit, wherein the second test sub-circuit is configured to connect the shift register of the second test sub-circuit to a scan chain of the system-on-chip device in the scan test mode of the test circuit, wherein the third test sub-circuit is configured to connect the shift register of the third test sub-circuit to the scan chain of the system-on-chip device in the scan test mode of the test circuit.

30 In example 18, the subject matter of one of the examples 1-17 can optionally include the input/output interface circuit comprising a plurality of input/output channel circuits connected to the first circuit and connected to the second circuit.

In example 19, the subject matter of example 18 can optionally include each input/output channel circuit of the plurality of input/output channel circuits comprising a memory circuit configured to be written and to be read at different clock frequencies.

- 5 In example 20, the subject matter of example 19 can optionally include each memory circuit comprising a first-in-first-out circuit.

- In example 21, the subject matter of example 19 or 20 can optionally include each memory circuit comprises at least one of an asynchronous first-in first-out memory circuit and a
10 bubble generation first-in first-out memory circuit.

In example 22, the subject matter of one of the previous examples can optionally include the first circuit being a core circuit.

- 15 In example 23, the subject matter of one of the previous examples can optionally include the first circuit being a central processing unit.

In example 24, the subject matter of one of the previous examples can optionally include the second circuit being an uncore circuit.

- 20 In example 25, the subject matter of one of the previous examples can optionally include the first circuit, the second circuit, the input/output interface circuit and the test circuit being located on the same semiconductor die.

- 25 In example 26, the subject matter of one of the previous examples can optionally include the first circuit, the second circuit, the input/output interface circuit and the test circuit being digital circuits.

- Example 27 relates to a system-on-chip device comprising a first circuit, a second circuit, an
30 input/output interface circuit configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit and configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit, and a test circuit configured to provide a test input signal to the input/output interface circuit

and configured to receive a corresponding test output signal from the input/output interface circuit.

5 In example 28, the subject matter of example 27 can optionally include the test circuit being configured to provide the test input signal containing test input data at a write frequency and configured to read the test output signal at a read frequency, wherein the write frequency is higher than the read frequency in a first test mode.

10 In example 29, the subject matter of example 28 can optionally include the write frequency being lower than the read frequency in a second test mode.

Example 30 relates to a method for testing a system-on-chip device, the method comprising operating a first circuit in a first clock domain, operating a second circuit in a second clock domain, wherein a clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device, and at
15 least one of providing a test input signal to an input/output interface circuit and receiving a test output signal from the input/output interface circuit in a test mode of the system-on-chip device, wherein the input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding output signals to the second circuit and to receive input signals from the second circuit and provide
20 corresponding output signals to the first circuit.

In example 31, the subject matter of example 30 can optionally include the clock frequency of the second clock domain being higher than the clock frequency of the first clock domain
25 in a second operating state of the system-on-chip device.

In example 32, the subject matter of one of the examples 30 to 31 can optionally include providing a test input signal to the input/output interface circuit in an input test mode of the first test sub-circuit.
30

In example 33, the subject matter of one of the examples 30 to 32 can optionally include receiving a test output signal from the input/output interface circuit and providing an output test signal to an output of the system-on-chip device or a scan chain of the system-on-chip device in an output test mode.
35

In example 34, the subject matter of one of the examples 30 to 33 can optionally include connecting a shift register of the test circuit to a scan chain of the system-on-chip device in a scan test mode.

- 5 In example 35, the subject matter of example 34 can optionally include capturing output data of at least one of the first circuit, the second circuit and the input/output interface circuit by the shift register in a capture test mode.

- 10 In example 36, the subject matter of one of the examples 30 to 35 can optionally include providing predefined signals to at least one of the first circuit and the second circuit instead of the output signals of the input/output interface circuit in a sealing test mode in a test operation of at least one of the first circuit and the second circuit.

- 15 Example 37 relates to a method for testing a system-on-chip device, the method comprising providing a test input signal to the input/output interface circuit, wherein the input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding output signals to the second circuit and to receive input signals from the second circuit and provide corresponding output signals to the first circuit, and receiving a corresponding test output signal from the input/output interface circuit.

- 25 In example 38, the subject matter of example 37 can optionally include the test input signal containing test input data at a write frequency and receiving the test output signal comprises reading the test output signal at a read frequency, wherein the write frequency is higher than the read frequency in a first test mode.

In example 39, the subject matter of example 38 can optionally include the write frequency being lower than the read frequency in a second test mode.

- 30 Example 40 is a machine readable storage medium including program code, when executed, to cause a machine to perform the method of one of the examples 31 to 39.

- 35 Examples may further provide a computer program having a program code for performing one of the above methods, when the computer program is executed on a computer or processor. A person of skill in the art would readily recognize that steps of various above-

described methods may be performed by programmed computers. Herein, some examples are also intended to cover program storage devices, e.g., digital data storage media, which are machine or computer readable and encode machine-executable or computer-executable programs of instructions, wherein the instructions perform some or all of the acts of the
5 above-described methods. The program storage devices may be, e.g., digital memories, magnetic storage media such as magnetic disks and magnetic tapes, hard drives, or optically readable digital data storage media. The examples are also intended to cover computers programmed to perform the acts of the above-described methods or (field) programmable logic arrays ((F)PLAs) or (field) programmable gate arrays ((F)PGAs), programmed to perform
10 the acts of the above-described methods.

The description and drawings merely illustrate the principles of the disclosure. It will thus be appreciated that those skilled in the art will be able to devise various arrangements that, although not explicitly described or shown herein, embody the principles of the disclosure
15 and are included within its spirit and scope. Furthermore, all examples recited herein are principally intended expressly to be only for pedagogical purposes to aid the reader in understanding the principles of the disclosure and the concepts contributed by the inventor(s) to furthering the art, and are to be construed as being without limitation to such specifically recited examples and conditions. Moreover, all statements herein reciting principles, as-
20 pects, and examples of the disclosure, as well as specific examples thereof, are intended to encompass equivalents thereof.

Functional blocks denoted as “means for ...” (performing a certain function) shall be understood as functional blocks comprising circuitry that is configured to perform a certain func-
25 tion, respectively. Hence, a “means for s.th.” may as well be understood as a “means configured to or suited for something”. A means configured to perform a certain function does, hence, not imply that such means necessarily is performing the function (at a given time instant).

30 Functions of various elements shown in the figures, including any functional blocks labeled as “means”, “means for providing a sensor signal”, “means for generating a transmit signal.”, etc., may be provided through the use of dedicated hardware, such as “a signal provider”, “a signal processing unit”, “a processor”, “a controller”, etc. as well as hardware capable of executing software in association with appropriate software. Moreover, any enti-

ty described herein as “means”, may correspond to or be implemented as “one or more modules”, “one or more devices”, “one or more units”, etc. When provided by a processor, the functions may be provided by a single dedicated processor, by a single shared processor, or by a plurality of individual processors, some of which may be shared. Moreover, explicit
5 use of the term “processor” or “controller” should not be construed to refer exclusively to hardware capable of executing software, and may implicitly include, without limitation, digital signal processor (DSP) hardware, network processor, application specific integrated circuit (ASIC), field programmable gate array (FPGA), read only memory (ROM) for storing software, random access memory (RAM), and non-volatile storage. Other hardware,
10 conventional and/or custom, may also be included.

It should be appreciated by those skilled in the art that any block diagrams herein represent conceptual views of illustrative circuitry embodying the principles of the disclosure. Similarly, it will be appreciated that any flow charts, flow diagrams, state transition diagrams,
15 pseudo code, and the like represent various processes which may be substantially represented in computer readable medium and so executed by a computer or processor, whether or not such computer or processor is explicitly shown.

Furthermore, the following claims are hereby incorporated into the Detailed Description,
20 where each claim may stand on its own as a separate example. While each claim may stand on its own as a separate example, it is to be noted that - although a dependent claim may refer in the claims to a specific combination with one or more other claims - other examples may also include a combination of the dependent claim with the subject matter of each other dependent or independent claim. Such combinations are proposed herein unless it is stated
25 that a specific combination is not intended. Furthermore, it is intended to include also features of a claim to any other independent claim even if this claim is not directly made dependent to the independent claim.

It is further to be noted that methods disclosed in the specification or in the claims may be
30 implemented by a device having means for performing each of the respective acts of these methods.

Further, it is to be understood that the disclosure of multiple acts or functions disclosed in the specification or claims may not be construed as to be within the specific order. There-

fore, the disclosure of multiple acts or functions will not limit these to a particular order unless such acts or functions are not interchangeable for technical reasons. Furthermore, in some examples a single act may include or may be broken into multiple sub acts. Such sub acts may be included and part of the disclosure of this single act unless explicitly excluded.

Claims

1. A system-on-chip device comprising:
 - 5 A first circuit configured to operate in a first clock domain;
 - A second circuit configured to operate in a second clock domain, wherein a clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device;
 - 10 An input/output interface circuit configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit and configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit; and
 - 15 A test circuit connected to the input/output interface circuit and configured to at least one of providing a test input signal to the input/output interface circuit and receiving a test output signal from the input/output interface circuit in a test mode of the system-on-chip device.
- 20 2. The system-on-chip device according to claim 1, wherein the clock frequency of the second clock domain is higher than the clock frequency of the first clock domain in a second operating state of the system-on-chip device.
3. The system-on-chip device according to claim 1, wherein the test circuit comprises a
25 first test sub-circuit connected to the input/output interface circuit.
4. The system-on-chip device according to claim 3, wherein the first test sub-circuit is configured to provide a test input signal to the input/output interface circuit in an input test mode of the first test sub-circuit.
- 30 5. The system-on-chip device according to claim 4, wherein the test input signal is a random or pseudo-random bit sequence.

6. The system-on-chip device according to claim 3, wherein the first test sub-circuit is configured to receive a test output signal from the input/output interface circuit and provide an output test signal to an output of the system-on-chip device or a scan chain of the system-on-chip device in an output test mode of the first test sub-circuit.

5

7. The system-on-chip device according to claim 6, wherein the first test sub-circuit is configured as a multiple-input signature register circuit in the output test mode or comprises a multiple-input signature register circuit, wherein the multiple-input signature register circuit is configured to generate the output test signal of the first test sub-circuit indicating a signature of the test output signal of the input/output interface circuit in the output test mode.

10

8. The system-on-chip device according to claim 3, wherein the first test sub-circuit comprises a shift register or the first test sub-circuit is configured as a shift register in a scan test mode of the first test sub-circuit, wherein the first test sub-circuit is configured to connect the shift register to a scan chain of the system-on-chip device in the scan test mode.

15

9. The system-on-chip device according to claim 8, wherein the first test sub-circuit is configured to capture output data of at least one of the first circuit, the second circuit and the input/output interface circuit by the shift register in a capture test mode of the first test sub-circuit.

20

10. The system-on-chip device according to claim 3, wherein the test circuit is configured to provide predefined signals to at least one of the first circuit and the second circuit instead of the output signals of the input/output interface circuit in a sealing test mode of the test circuit in a test operation of at least one of the first circuit and the second circuit.

25

11. The system-on-chip device according to claim 10, wherein the predefined signals are constant signals in the sealing test mode of the test circuit.

30

12. The system-on-chip device according to claim 3, wherein the first test sub-circuit comprises a build-in logic block observer circuit.

13. The system-on-chip device according to claim 3, wherein the test circuit comprises a second test sub-circuit connected to the input/output interface circuit.

14. The system-on-chip device according to claim 13, wherein the second test sub-circuit is configured to provide a test input signal to the input/output interface circuit in an inbound test mode of the test circuit, wherein the first test sub-circuit is configured to receive a test output signal from the input/output interface circuit based on the test input signal provided by the second test sub-circuit in the inbound test mode of the test circuit.

15. The system-on-chip device according to claim 13, wherein the test circuit comprises a third test sub-circuit connected to the input/output interface circuit.

16. The system-on-chip device according to claim 15, wherein the first test sub-circuit is configured to provide a test input signal to the input/output interface circuit in an outbound test mode of the test circuit, wherein the third test sub-circuit is configured to receive a test output signal from the input/output interface circuit based on the test input signal provided by the first test sub-circuit in the outbound test mode of the test circuit.

17. The system-on-chip device according to claim 15, wherein the second test sub-circuit comprises a shift register or the second test sub-circuit is configured as a shift register in a scan test mode of the test circuit, wherein the third test sub-circuit comprises a shift register or the third test sub-circuit is configured as a shift register in the scan test mode of the test circuit, wherein the second test sub-circuit is configured to connect the shift register of the second test sub-circuit to a scan chain of the system-on-chip device in the scan test mode of the test circuit, wherein the third test sub-circuit is configured to connect the shift register of the third test sub-circuit to the scan chain of the system-on-chip device in the scan test mode of the test circuit.

18. The system-on-chip device according to claim 1, wherein the input/output interface circuit comprises a plurality of input/output channel circuits connected to the first circuit and connected to the second circuit.

19. The system-on-chip device according to claim 18, wherein each input/output channel circuit of the plurality of input/output channel circuits comprises a memory circuit configured to be written and to be read at different clock frequencies.

5 20. A system-on-chip device comprising:

A first circuit;

A second circuit;

10

An input/output interface circuit configured to receive input signals from the first circuit and provide corresponding output signals to the second circuit and configured to receive input signals from the second circuit and provide corresponding output signals to the first circuit; and

15

A test circuit configured to provide a test input signal to the input/output interface circuit and configured to receive a corresponding test output signal from the input/output interface circuit.

20 21. The system-on-chip device according to claim 20, wherein the test circuit is configured to provide the test input signal containing test input data at a write frequency and configured to read the test output signal at a read frequency, wherein the write frequency is higher than the read frequency in a first test mode.

25 22. The system-on-chip device according to claim 21, wherein the write frequency is lower than the read frequency in a second test mode.

23. A method for testing a system-on-chip device, the method comprising:

30 Operating a first circuit in a first clock domain;

Operating a second circuit in a second clock domain, wherein a clock frequency of the first clock domain is higher than a clock frequency of the second clock domain in a first operating state of the system-on-chip device;

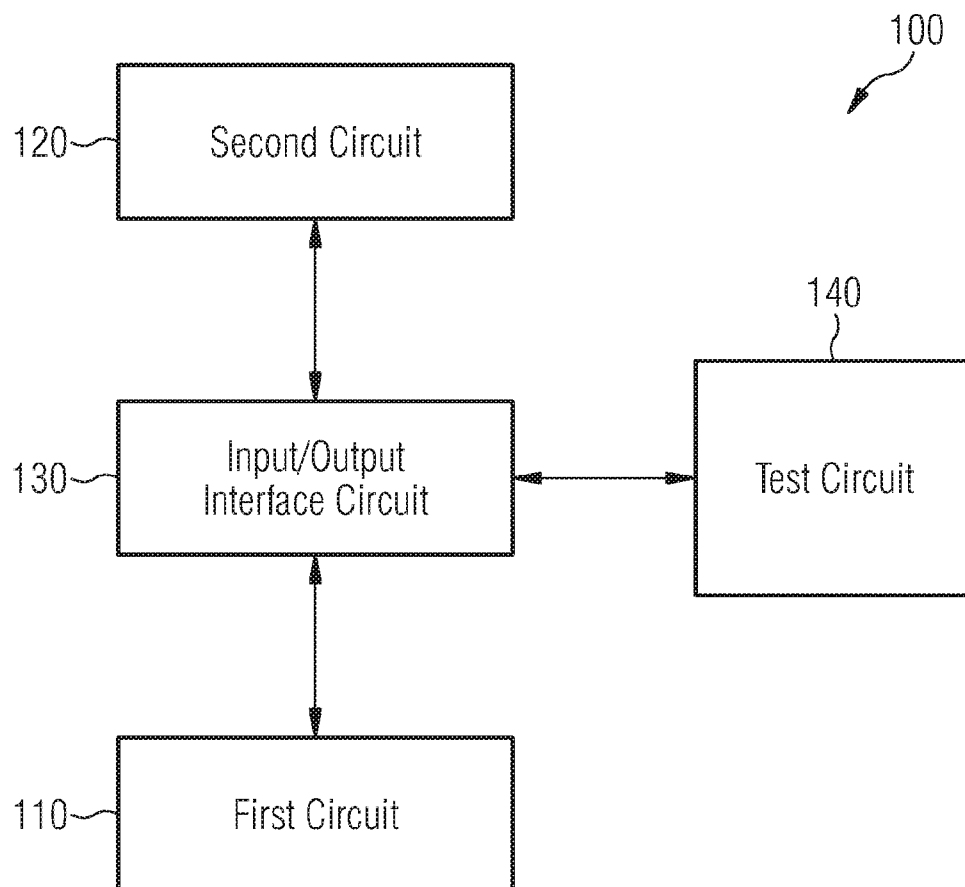
35

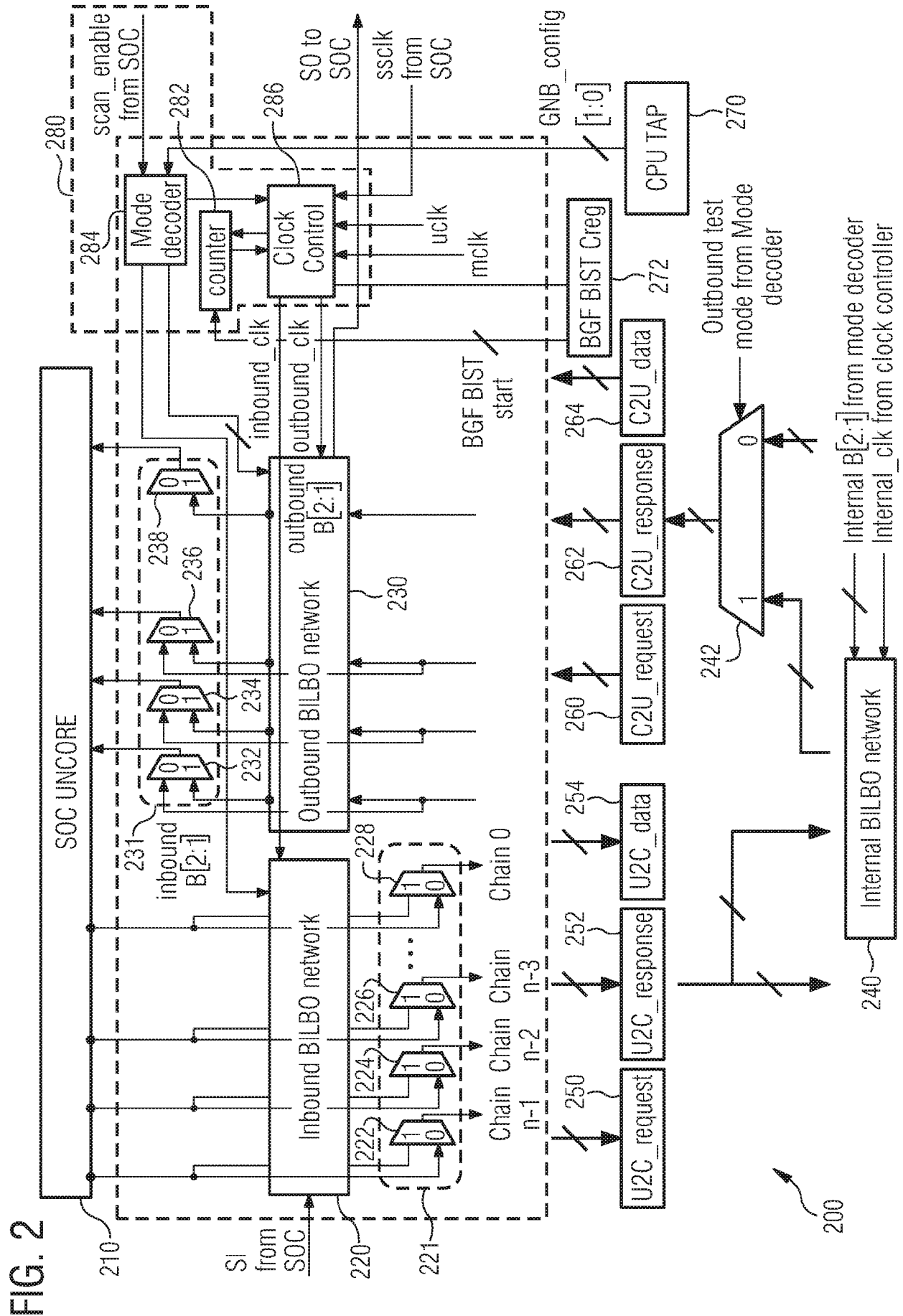
At least one of providing a test input signal to an input/output interface circuit and receiving a test output signal from the input/output interface circuit in a test mode of the system-on-chip device, wherein the input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding output signals to the second circuit and to receive input signals from the second circuit and provide corresponding output signals to the first circuit.

24. A method for testing a system-on-chip device, the method comprising:
- 10 Providing a test input signal to the input/output interface circuit, wherein the input/output interface circuit is connected to the first circuit and the second circuit to receive input signals from the first circuit and provide corresponding output signals to the second circuit and to receive input signals from the second circuit and provide corresponding output signals to the first circuit; and
- 15 Receiving a corresponding test output signal from the input/output interface circuit.

25. A machine readable storage medium including program code, when executed, to cause a machine to perform the method according to claim 23.
- 20

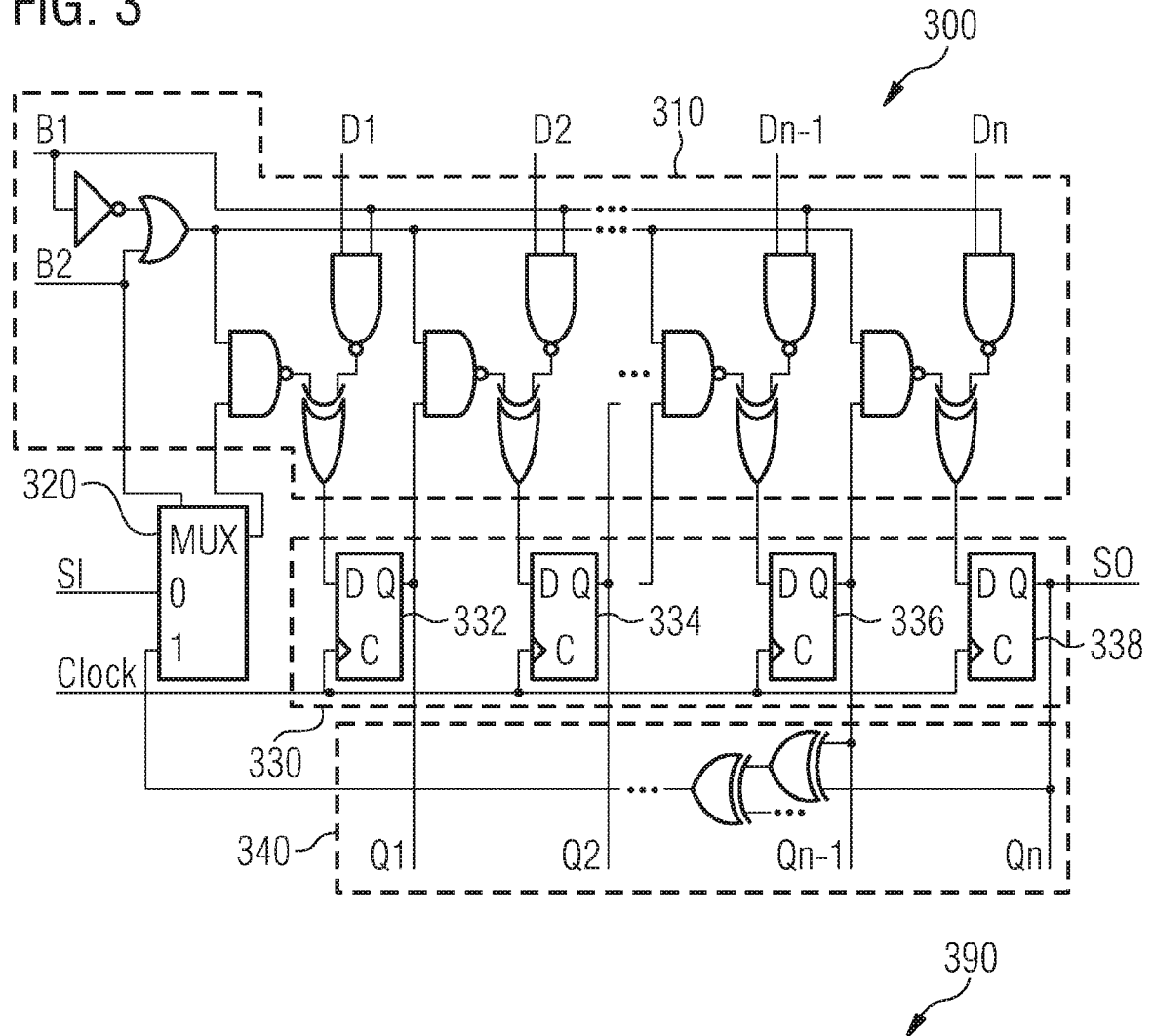
FIG. 1





3/12

FIG. 3



B1	B2	Mode	Usage
0	0	Serial scan shift mode	Allow shifting data from SI through all the flops to the SO
0	1	LFSR mode	Used as random pattern generator
1	0	D-FF mode	Capture data from D pin into D-FF
1	1	MISR mode	Used to compact the data response into one signature

FIG. 4

4/12

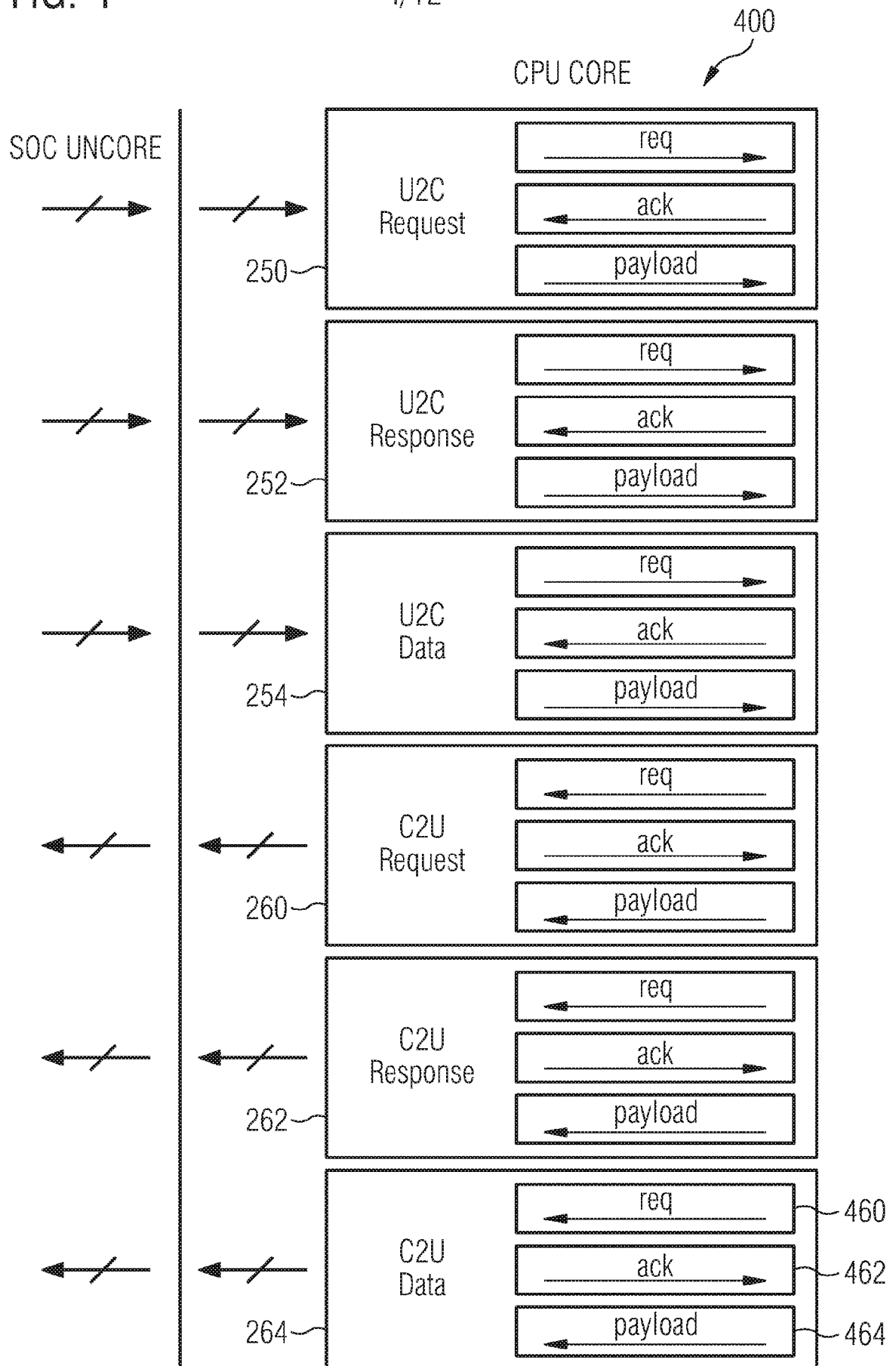


FIG. 5

500

TAP_BILBO_CFG[1:0]	Modes	Inbound BILBO network			Outbound BILBO network			Internal BILBO network		
		B1	B2	CLK	B1	B2	CLK	B1	B2	CLK
		X	X	off	X	X	off	X	X	off
2'b00	Function Mode									
2'b01	SOC ATPG mode or debug mode	~(se or TAP shift_dr)	0	ssclk /tck	~(se or TAP shift_dr)	0	ssclk /tck	~(TAP shift_dr)	0	tck
2'b10	inbound BGF BIST	0	1	uclk	X	X	off	1	1	mclk
2'b11	outbound BGF BIST	X	X	off	1	1	uclk	0	1	mclk

FIG. 6A

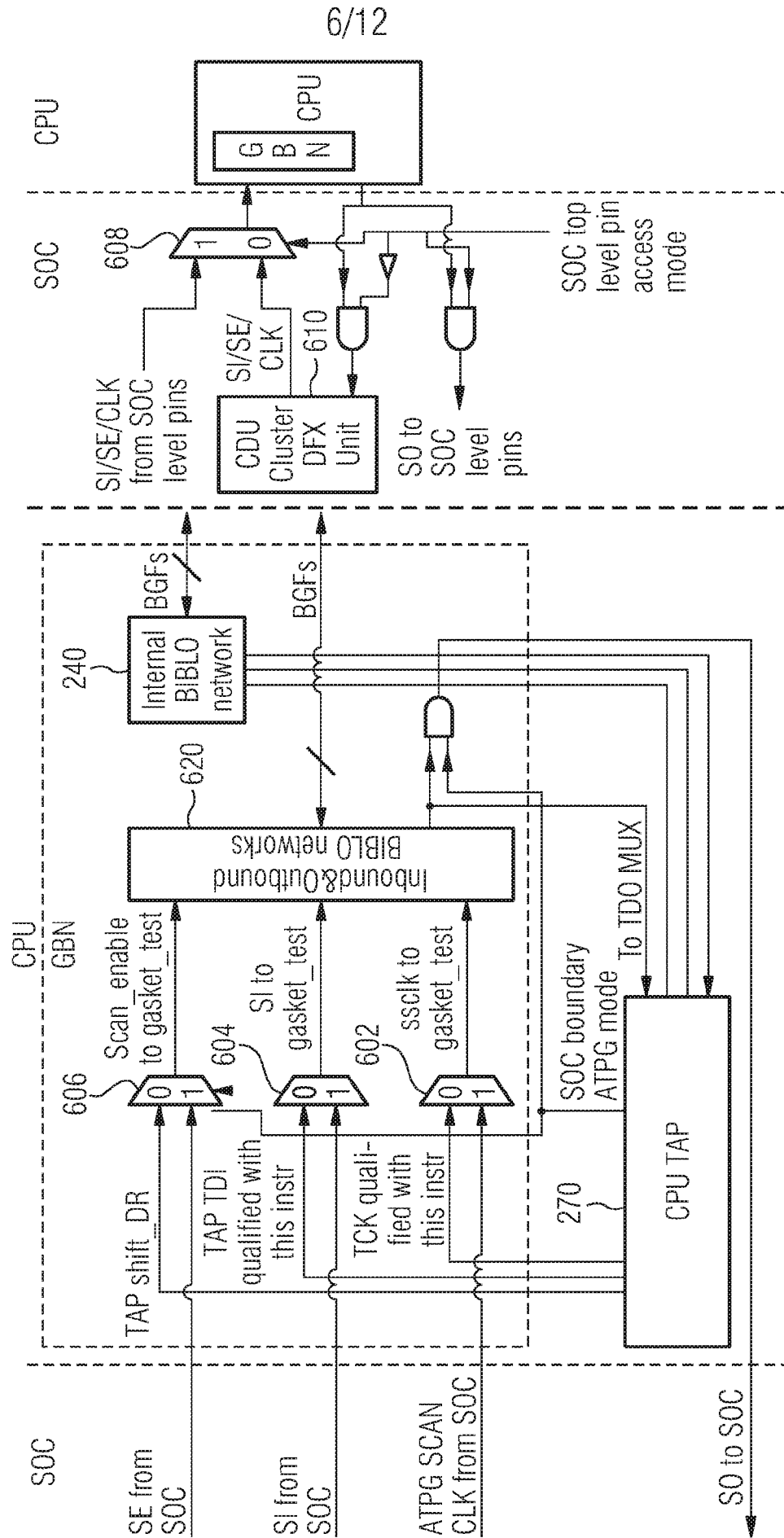
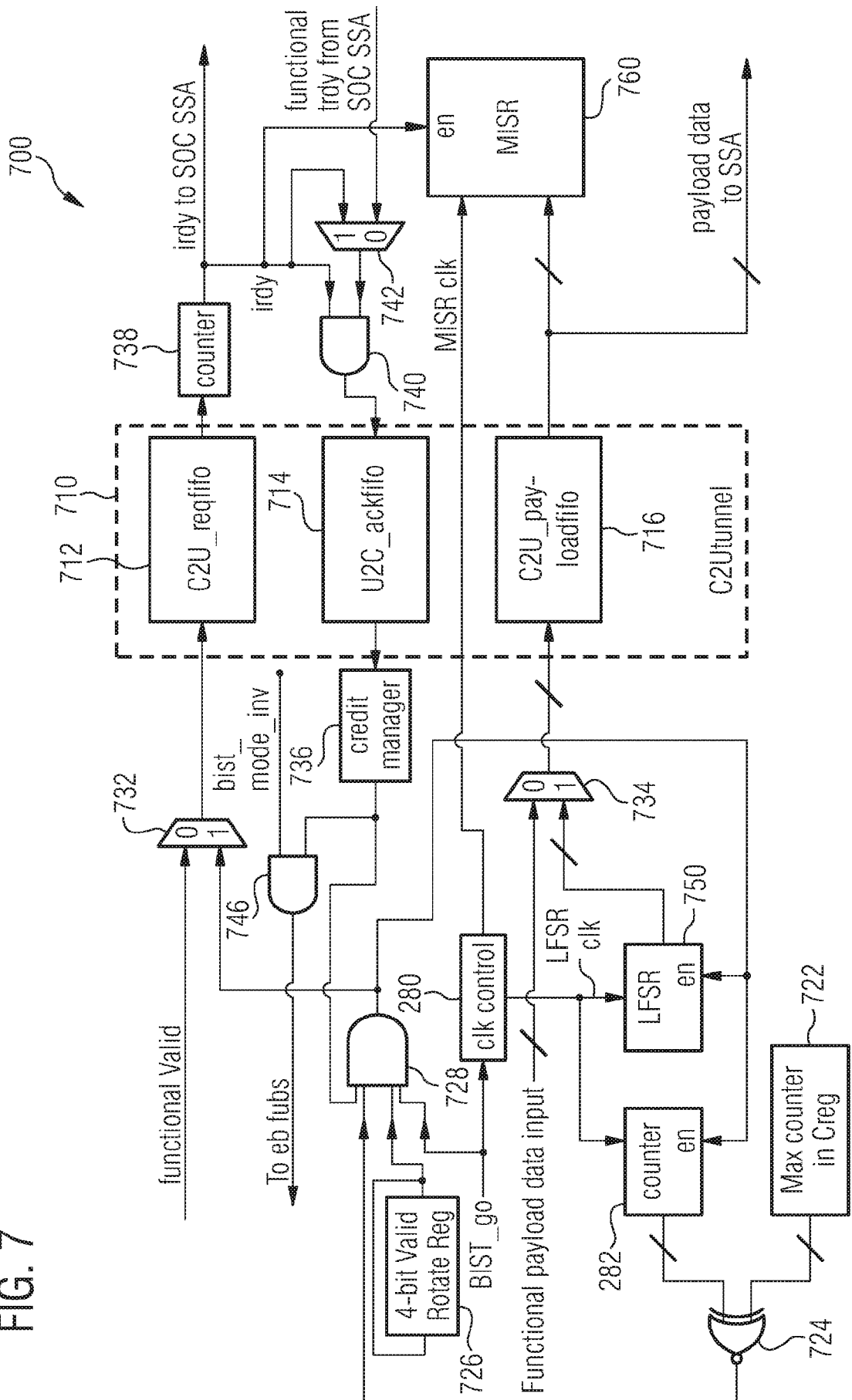


FIG. 6B

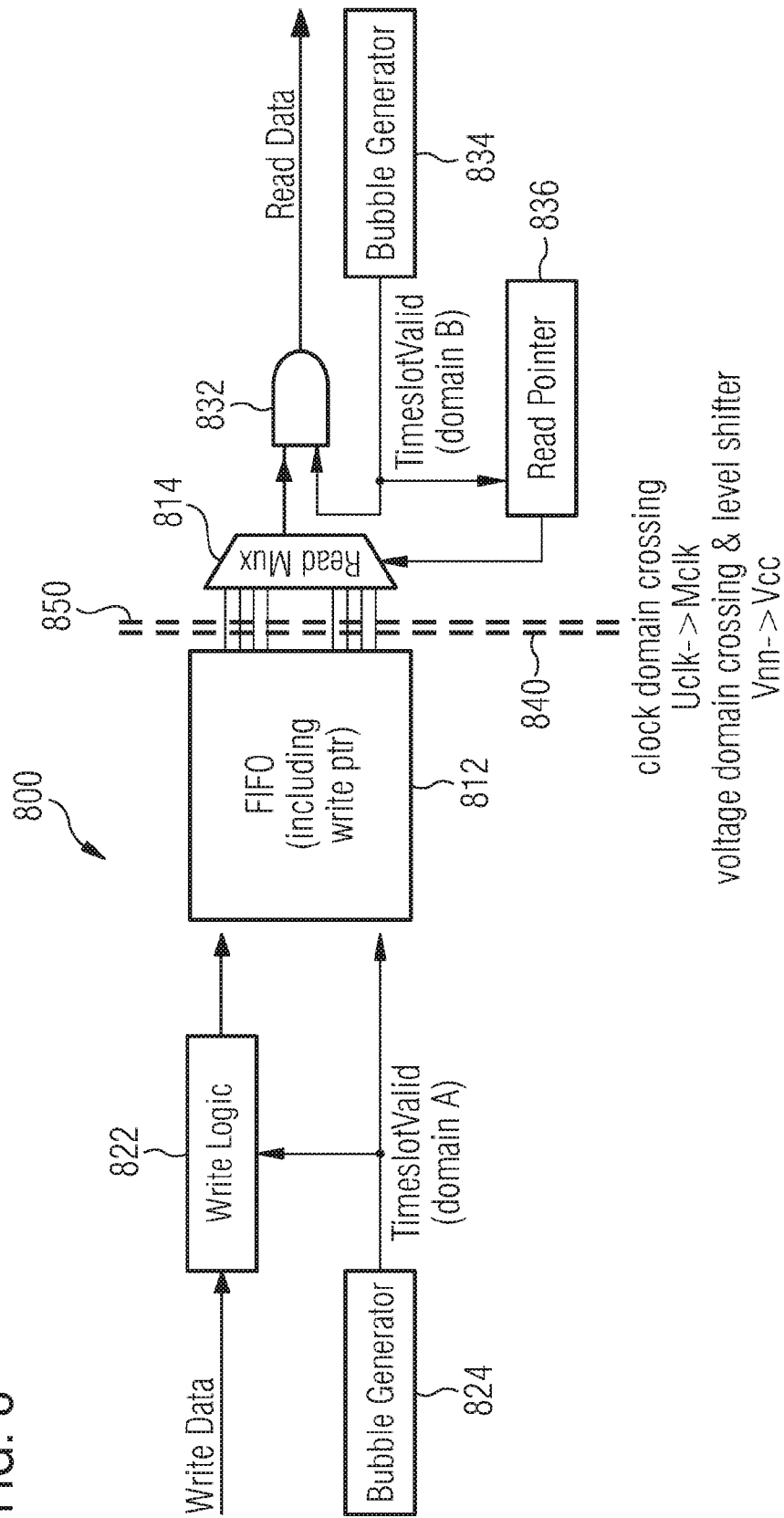
7/12

FIG. 7

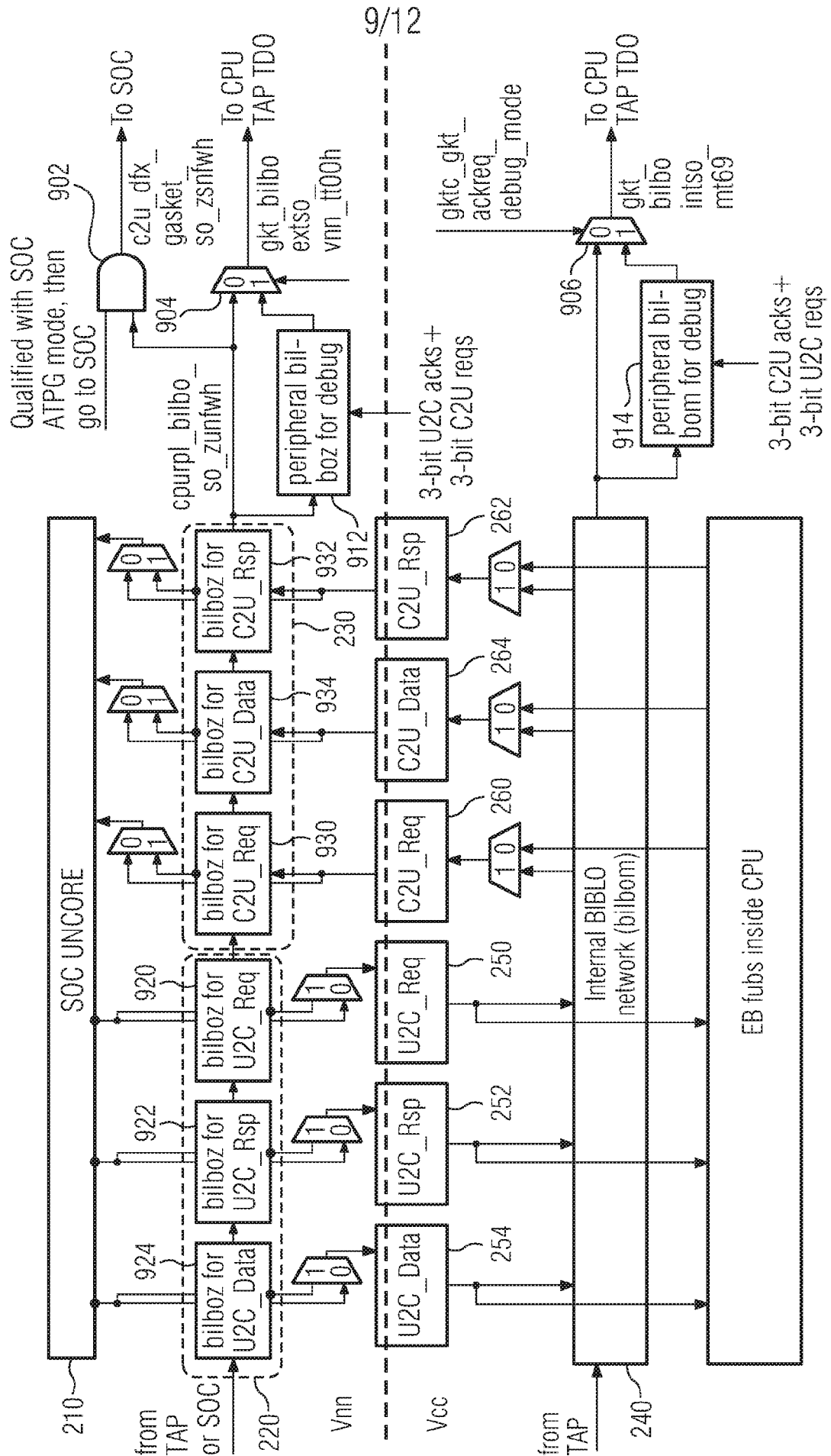


8/12

FIG. 8

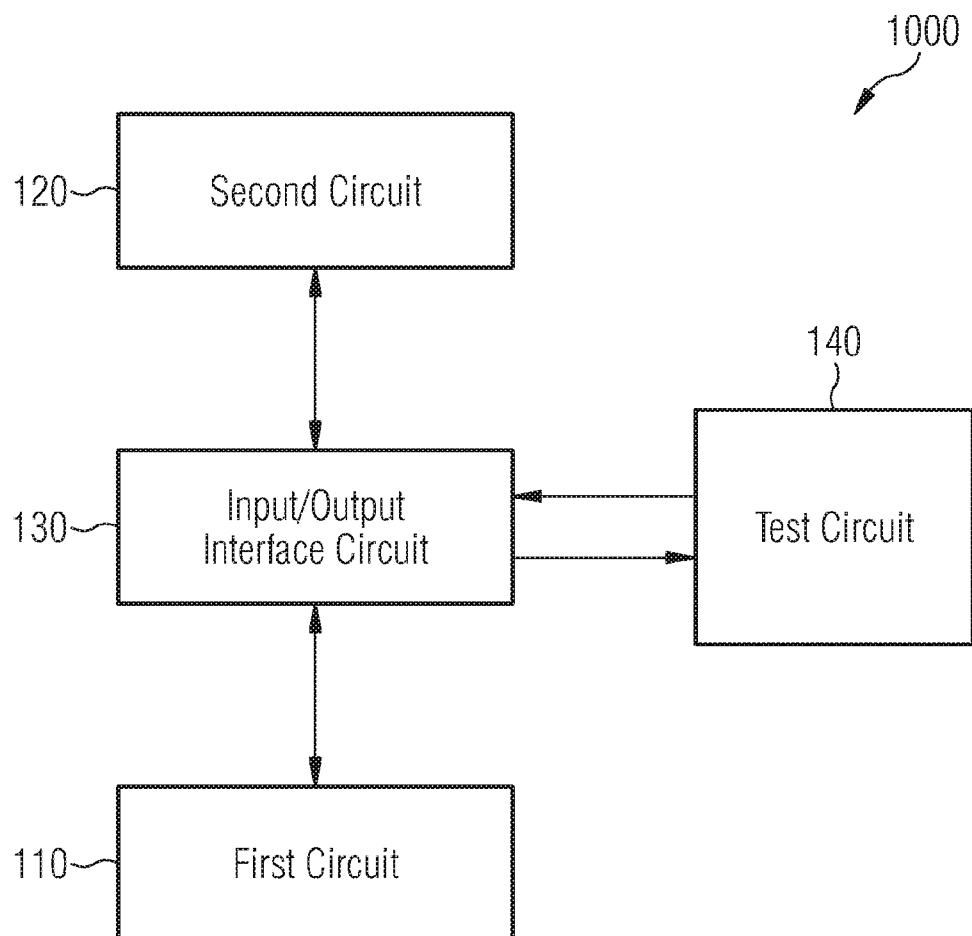


၈၆



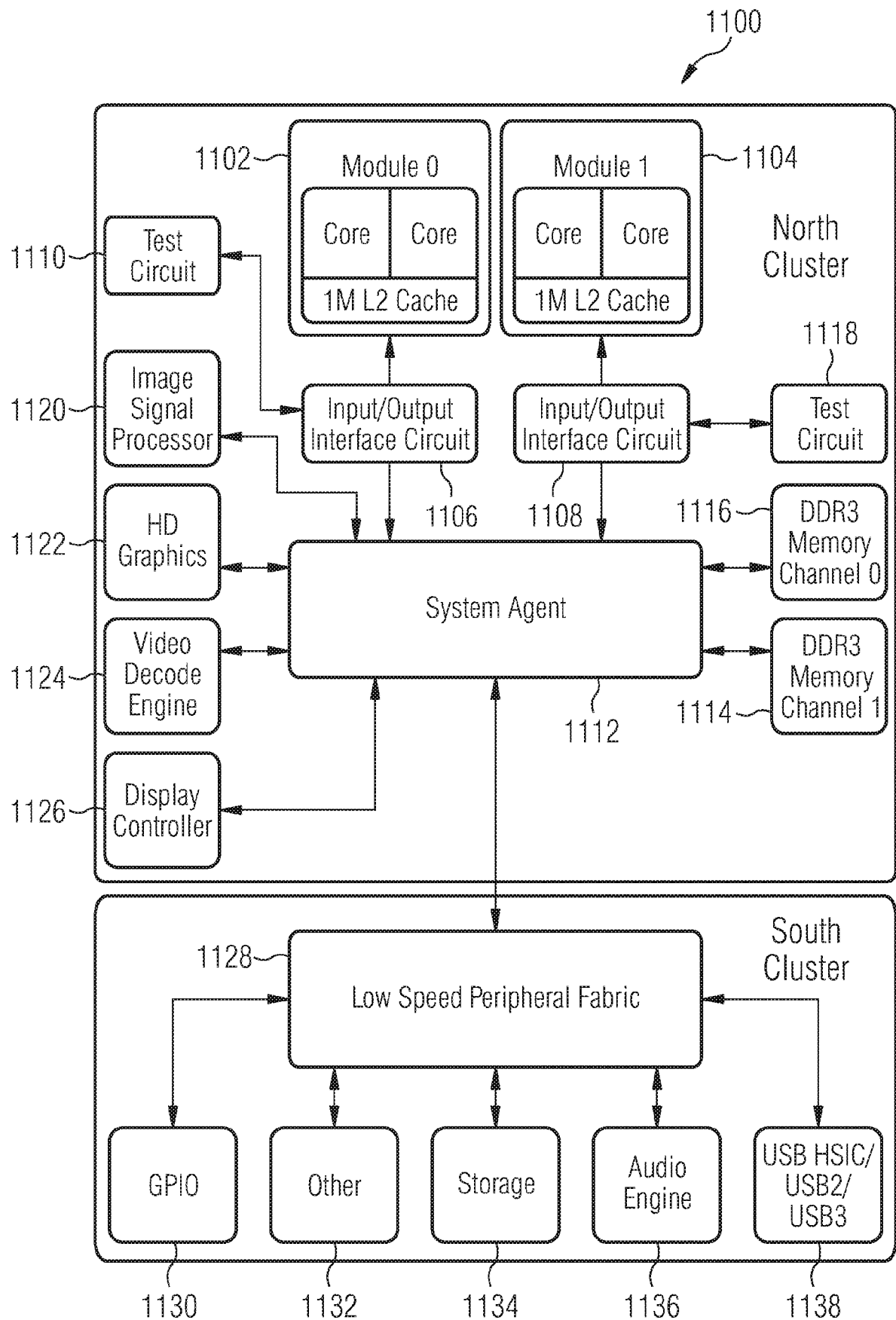
10/12

FIG. 10



11/12

FIG. 11



12/12

FIG. 12

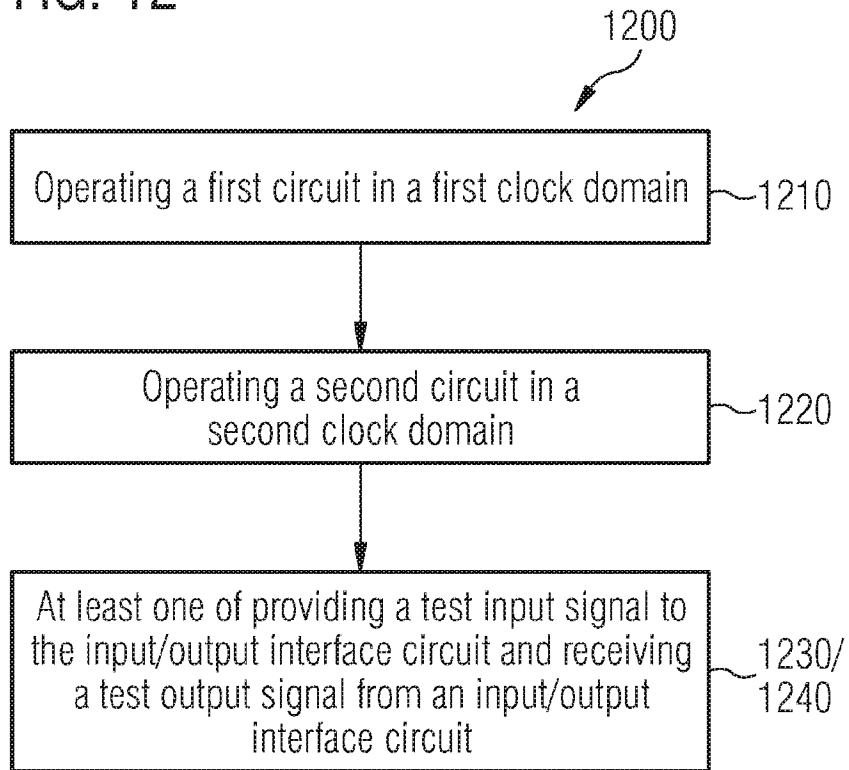
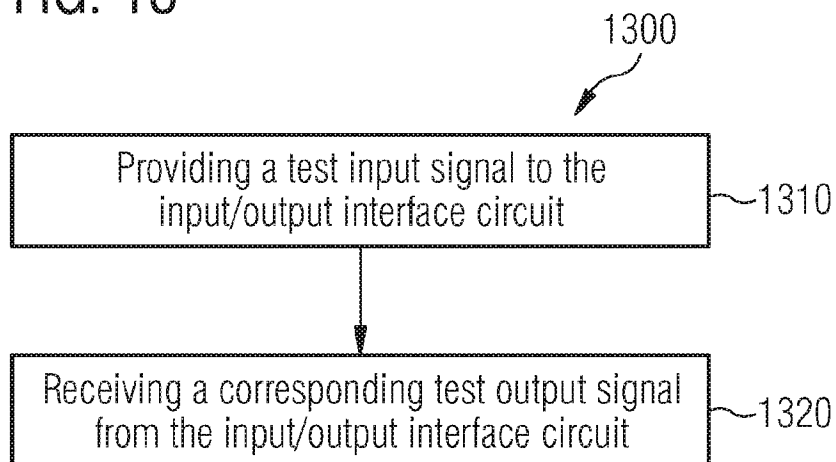


FIG. 13



INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/023866**A. CLASSIFICATION OF SUBJECT MATTER****G01R 31/317(2006.01)i, G01R 31/3181(2006.01)i, G01R 31/3183(2006.01)i, G01R 31/3187(2006.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G01R 31/317; G06F 11/26; G06F 11/25; G01R 31/3177; G01R 31/28; G01R 31/00; G01R 31/3181; G01R 31/3183; G01R 31/3187

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & Keywords: test, system, chip, circuit, clock, interface, first, second

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0453106 B1 (STEPHENS et al.) 12 June 1996 See column 4, lines 34-36, claim 1 and figures 1-3.	20-22, 24
A		1-19, 23, 25
A	US 7653852 B2 (MAEHARA) 26 January 2010 See abstract, claims 1-18 and figure 3.	1-25
A	US 7024328 B2 (HOLT et al.) 04 April 2006 See abstract, claims 1-20 and figures 2-4.	1-25
A	US 2009-0019328 A1 (VAN DE LOGT) 15 January 2009 See abstract, claims 1-9 and figure 1.	1-25
A	US 2005-0204233 A1 (SHIN et al.) 15 September 2005 See abstract, claims 1-23 and figure 3.	1-25



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

19 December 2016 (19.12.2016)

Date of mailing of the international search report

20 December 2016 (20.12.2016)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

KIM, Yeon Kyung

Telephone No. +82-42-481-3325



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/023866

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0453106 B1	12/06/1996	AT 139355 T DE 69120142 T2 EP 0453106 A1 GB 2245105 A GB 2245105 B JP 07-092229 A JP 2001-000042 U US 5134638 A	15/06/1996 10/10/1996 23/10/1991 18/12/1991 05/01/1994 07/04/1995 07/12/2001 28/07/1992
US 7653852 B2	26/01/2010	JP 2007-212339 A JP 5160039 B2 US 2007-0192660 A1	23/08/2007 13/03/2013 16/08/2007
US 7024328 B2	04/04/2006	US 2005-0165572 A1	28/07/2005
US 2009-0019328 A1	15/01/2009	CN 101395488 A EP 1994421 A2 JP 2009-528535 A WO 2007-099479 A2 WO 2007-099479 A3	25/03/2009 26/11/2008 06/08/2009 07/09/2007 13/12/2007
US 2005-0204233 A1	15/09/2005	CN 1661388 A JP 2005-241651 A KR 10-0594257 B1 KR 10-2005-0087270 A US 2008-0313515 A1 US 7421635 B2 US 7761763 B2	31/08/2005 08/09/2005 30/06/2006 31/08/2005 18/12/2008 02/09/2008 20/07/2010