



- (51) International Patent Classification:
G01R 31/317 (2006.01) *G01R 31/3185* (2006.01)
- (21) International Application Number:
PCT/US2017/027238
- (22) International Filing Date:
12 April 2017 (12.04.2017)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
15/154,266 13 May 2016 (13.05.2016) US
- (71) Applicant: ALTERA CORPORATION [US/US]; 101 Innovation Drive, San Jose, CA 95134 (US).
- (72) Inventor: LIEW, Weng, Hong; 32 Jalan Besar, Sungkai Perak, 35600 (MY).

- (74) Agent: TSAI, Jason; Treyz Law Group, P.C., 870 Market Street, Suite 984, San Francisco, CA 94102 (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, DJ, 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, KH, KN, KP, KR, KW, 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,

- (54) Title: EMBEDDED BUILT-IN SELF-TEST (BIST) CIRCUITRY FOR DIGITAL SIGNAL PROCESSOR (DSP) VALIDATION

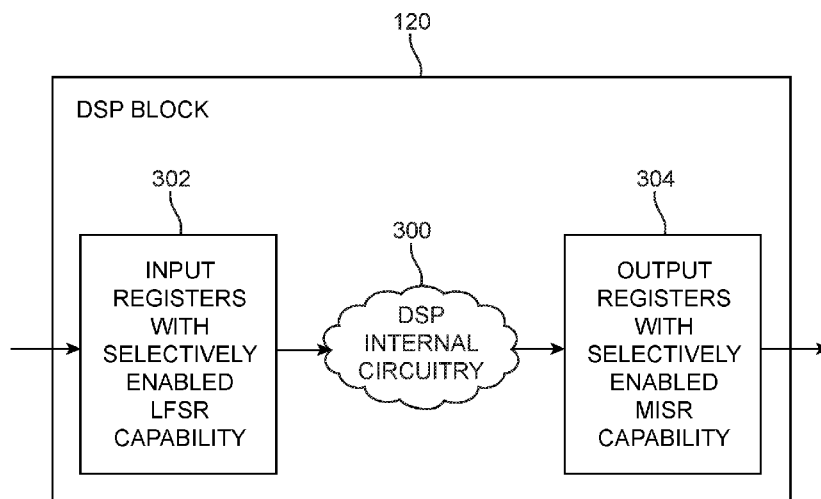


FIG. 3A

- (57) Abstract: Programmable integrated circuits with specialized processing blocks such as digital signal processing (DSP) blocks are provided. Each DSP block may include embedded built-in self-test circuitry implemented using existing input registers and output registers in the DSP blocks. The input registers may be selectively coupled in a loop to serve as a linear feedback shift register (LFSR). The output registers may be selectively coupled in a chain to serve as a multiple input signature register (MISR). Configured in this way, the LFSR and the MISR circuits of the DSP blocks are not implemented using soft logic and can therefore easily meet performance criteria.

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))*

EMBEDDED BUILT-IN SELF-TEST (BIST) CIRCUITRY FOR
DIGITAL SIGNAL PROCESSOR (DSP) VALIDATION

This application claims priority to U.S. patent application No. 15/154,266, filed on May 13, 2016, which is hereby incorporated by reference herein in its entirety.

Background

This invention relates to testing integrated circuits and more particularly, to testing integrated circuit using built-in self-test circuitry.

5 Consider a configurable device such as a programmable logic device (PLD) as one example of an integrated circuit. As applications for which configurable devices are used increase in complexity, it

has become more common to include specialized processing blocks in configurable devices. Such types of specialized processing blocks—commonly referred to as digital signal processing blocks or DSP blocks—may include a

5 concentration of circuitry that has been partly or fully hardwired to perform one or more specific tasks, such as a logical or a mathematical operation.

To ensure satisfactory operation of an integrated circuit that contains DSP blocks, the DSP
10 blocks are generally tested during debug and design operations. Testing may also be performed during manufacturing.

It can be cumbersome to perform high-speed testing using only external test equipment. It is
15 therefore often desirable to include internal test support circuitry on an integrated circuit to facilitate testing of the specialized processing blocks. Because the internal test circuitry is incorporated into the integrated circuit that is to be tested, this type of test
20 circuitry is sometimes referred to as built-in self-test (BIST) circuitry.

Conventionally, performance validation using BIST circuitry has involved using available soft logic resource on a programmable logic device to form a linear
25 feedback shift register at the input of a DSP block and a multiple input signature register at the output of the DSP block. Implemented using soft logic, the linear feedback shift register and the multiple input signature register present two additional logic circuits per DSP block that
30 need to be modeled and verified during the integrated circuit design flow. The timing analysis that has to be performed on these two circuits can be fairly time

consuming and is highly dependent on the timing model accuracy, which has become more unpredictable at smaller process nodes.

It is within this context that the embodiments
5 described herein arise.

Summary

This relates generally to integrated circuits and more particularly, to integrated circuits with built-
10 in self-test (BIST) circuitry. In accordance with some embodiments, a programmable integrated circuit die may be provided with programmable soft logic, configuration memory circuitry, and specialized processing blocks sometimes referred to as digital signal processing (DSP)
15 blocks. In particular, a DSP block may include embedded BIST circuitry that is not implemented using soft logic.

The embedded DSP BIST circuitry may be implemented using existing input registers and output registers within the DSP block. The input registers may
20 be selectively enabled to operate as a linear feedback shift register (LFSR), whereas the output registers may be selectively enabled to operate as a multiple input signature register (MISR). The input registers may be placed in an LFSR mode by selectively asserting a first
25 enable signal, whereas the output registers may be placed in an MISR mode by selectively asserting a second enable signal. The LFSR can include multiple multiplexers and a logic XOR gate coupled to the input registers in a loop. The MISR can include multiple logic XOR gates and
30 multiplexers coupled in a chain.

To perform validation on the DSP block, a seed BIST value may be fed into the input registers while the

first enable signal is deasserted. After latching the seed value, both the first and second enable signals may be simultaneously asserted. After assertion of the first and second enable signals, the BIST circuitry may wait for
5 a predetermined number of clock cycles before comparing a signature generated by the output registers with a pre-computed signature. If the signatures match, the operating frequency of the DSP block can be increased and validation continues. If the signatures are mismatched,
10 validation is complete.

Further features of the invention, its nature and various advantages will be more apparent from the accompanying drawings and following detailed description.

Brief Description of the Drawings

FIG. 1 is a diagram of an illustrative programmable integrated circuit in accordance with an
15 embodiment.

FIG. 2 is a diagram of a digital signal processing block with a separate test pattern generator and a separate output response analyzer that are implemented using soft logic.

20 FIG. 3A is a diagram of an illustrative specialized processing block with embedded selectively enabled linear feedback shift register (LFSR) circuit and multiple input signature register (MISR) circuit in accordance with an embodiment.

25 FIG. 3B is a circuit diagram of a selectively enabled linear feedback shift register circuit implemented using input registers within a specialized processing block in accordance with an embodiment.

FIG. 3C is a circuit diagram of a selectively enabled multiple input signature register circuit implemented using output registers within a specialized processing block in accordance with an embodiment.

5 FIG. 4 is a diagram of an integrated circuit design system that may be used to design integrated circuits in accordance with an embodiment.

FIG. 5 is a diagram of illustrative computer-aided design (CAD) tools that may be used in a circuit
10 design system in accordance with an embodiment.

FIG. 6 is a flow chart of illustrative steps for designing an integrated circuit in accordance with an embodiment.

FIG. 7 is a flow chart of illustrative steps for
15 performing validation on a specialized processing block with embedded built-in self-test circuitry in accordance with an embodiment.

Detailed Description

20 Embodiments of the present invention relate to integrated circuits and more particularly, to ways of providing built-in self-test optimization on programmable integrated circuits.

As described above in the Background section,
25 conventional methods for performing built-in self-test operations for digital signal processing (DSP) blocks in a programmable integrated circuit typically rely on using soft logic to implement an additional linear feedback shift register and an additional multiple input shift
30 register for each DSP block. The timing analysis that needs to be performed on these additional register circuits can be invariably time consuming.

In accordance with an embodiment, programmable integrated circuits may be provided with specialized processing blocks having embedded built-in self-test (BIST) circuitry that is not implemented using soft logic.

5 In particular, the embedded BIST circuitry may leverage any existing input and output registers within each specialized processing block and can be selectively enabled to perform BIST functions during validation or selectively bypassed during normal user modes. The
10 embedded BIST circuitry within each specialized processing blocks is specifically designed to have sufficient performance to carry out any desired functionality of that specialized processing block, so the time previously needed to ensure/verify BIST performance can be greatly
15 reduced.

It will be recognized by one skilled in the art, that the present exemplary embodiments may be practiced without some or all of these specific details. In other instances, well-known operations have not been described
20 in detail in order not to unnecessarily obscure the present embodiments.

An illustrative embodiment of a programmable integrated circuit such as programmable logic device (PLD) 100 that may be configured to implement a circuit design
25 is shown in FIG. 1. As shown in FIG. 1, the programmable logic device (PLD) may include a two-dimensional array of functional blocks, including logic array blocks (LABs) 110 and other functional blocks, such as random access memory (RAM) blocks 130 and digital signal processing (DSP)
30 blocks 120, for example. Functional blocks such as LABs 110 may include smaller programmable regions (e.g., logic elements, configurable logic blocks, or adaptive logic

modules) that receive input signals and perform custom functions on the input signals to produce output signals.

Programmable logic device 100 may contain programmable memory elements. Memory elements may be
5 loaded with configuration data (also called programming data) using input-output elements (IOEs) 102. Once loaded, the memory elements each provide a corresponding static control signal that controls the operation of an associated functional block (e.g., LABs 110, DSP 120, RAM
10 130, or input-output elements 102).

In a typical scenario, the outputs of the loaded memory elements are applied to the gates of metal-oxide-semiconductor transistors in a functional block to turn certain transistors on or off and thereby configure the
15 logic in the functional block including the routing paths. Programmable logic circuit elements that may be controlled in this way include parts of multiplexers (e.g., multiplexers used for forming routing paths in interconnect circuits), look-up tables, logic arrays, AND,
20 OR, NAND, and NOR logic gates, pass gates, etc.

The memory elements may use any suitable volatile and/or non-volatile memory structures such as random-access-memory (RAM) cells, fuses, antifuses, programmable read-only-memory memory cells, mask-
25 programmed and laser-programmed structures, combinations of these structures, etc. Because the memory elements are loaded with configuration data during programming, the memory elements are sometimes referred to as configuration memory, configuration RAM (CRAM), or programmable memory
30 elements.

In addition, the programmable logic device may have input-output elements (IOEs) 102 for driving signals

off of PLD and for receiving signals from other devices. Input-output elements 102 may include parallel input-output circuitry, serial data transceiver circuitry, differential receiver and transmitter circuitry, or other
5 circuitry used to connect one integrated circuit to another integrated circuit.

The PLD may also include programmable interconnect circuitry in the form of vertical routing channels 140 (i.e., interconnects formed along a vertical
10 axis of PLD 100) and horizontal routing channels 150 (i.e., interconnects formed along a horizontal axis of PLD 100), each routing channel including at least one track to route at least one wire. If desired, the interconnect circuitry may include pipeline elements, and the contents
15 stored in these pipeline elements may be accessed during operation. For example, a programming circuit may provide read and write access to a pipeline element.

Note that other routing topologies, besides the topology of the interconnect circuitry depicted in FIG. 1,
20 are intended to be included within the scope of the present invention. For example, the routing topology may include wires that travel diagonally or that travel horizontally and vertically along different parts of their extent as well as wires that are perpendicular to the
25 device plane in the case of three dimensional integrated circuits, and the driver of a wire may be located at a different point than one end of a wire. The routing topology may include global wires that span substantially all of PLD 100, fractional global wires such as wires that
30 span part of PLD 100, staggered wires of a particular length, smaller local wires, or any other suitable interconnection resource arrangement.

If desired, programmable logic device (PLD) 100 may be configured to implement a custom circuit design. For example, the configuration RAM may be programmed such that LABs 110, DSP 120, and RAM 130, programmable
5 interconnect circuitry (i.e., vertical channels 140 and horizontal channels 150), and the input-output elements 102 form the circuit design implementation.

Digital signal processor (DSP) blocks 120—sometimes referred to as “specialized processing
10 blocks”—may include a concentration of circuitry that has been partially or fully hardwired to perform one or more specific tasks, such as a logical or a mathematical operation. A DSP block may also contain one or more specialized structures, such as an array of configurable
15 memory elements. Examples of structures that are commonly implemented in such DSP blocks include: multipliers, adders, accumulators, arithmetic logic units (ALUs), barrel-shifters, various memory elements (such as FIFO/LIFO/SIPO/RAM/ROM/CAM blocks and register files), logic
20 AND/NAND/OR/NOR arrays, etc., or combinations thereof. Storage elements with a DSP may serve different purposes as configuration memory. For instance, storage elements within a DSP block may store coefficients for implementing FIR filters. Alternatively, storage elements within a DSP
25 block may be used to pipeline a critical path or to synchronize data before it is processed.

To ensure satisfactory operation of an integrated circuit that includes DSP blocks, the DSP blocks are generally tested during debug and design
30 operations. Testing may also be performed during manufacturing or during normal operation. Since testing using external test equipment is oftentimes infeasible,

internal test circuitry is oftentimes incorporated into the integrated circuit that is to be tested; this type of test circuitry is sometimes referred to as built-in self-test (BIST) circuitry.

5 Conventional BIST architectures for testing DSP blocks are satisfactory in certain situations, but can be inefficient when scaled to handle multiple DSP blocks. A system environment of the type that may be used during test operations is shown in FIG. 2. As shown in FIG. 2, a
10 discrete DSP block 200 is tested using associated BIST circuitry, which includes test pattern generator 202 and output response analyzer 204. Test pattern generator 202 is a linear feedback shift register (LFSR) that is built using soft reconfigurable logic on the integrated circuit
15 and that generates test patterns for inputting to DSP block 200. Output response analyzer 204 is a multiple input signature register (MISR) that is also built using soft reconfigurable logic on the integrated circuit and that generates a unique signature based on test outputs
20 received from DSP block 200.

 As described above in the Background section, any BIST circuitry that is implemented using soft logic needs to be optimized during the integrated circuit design flow, an extra step that can be fairly time consuming.
25 The extra time required for the timing analysis of test pattern generator 202 and output response analyzer 204 is further exacerbated for each DSP block that needs BIST circuitry. For example, if an integrated circuit includes 100 DSP blocks and if each of the 100 DSP blocks needs its
30 own dedicated BIST circuitry (i.e., a separate instantiation of test pattern generator 202 and output response analyzer 204 for each DSP block), the overall

duration of the timing analysis required to verify and close the timing of the extra BIST circuitry may be increased by more than two times, more than ten times, or more than 100 times, etc.

5 In accordance with an embodiment, the BIST circuitry may be embedded within the DSP block itself by leveraging any existing input and output registers inside the DSP block. FIG. 3A is a diagram showing a specialized processing block such as digital signal processing (DSP)
10 block 120 may include DSP internal (core) circuitry 300 and associated input and output register circuitry. In particular, DSP block 120 may include input register circuitry such as input registers 302 that can be selectively enabled to serve as a linear feedback shift
15 register (LFSR). Digital signal processing block 120 may also include output register circuitry such as output registers 304 that can be selectively enabled to serve as a multiple input signature register (MISR).

 During normal user operation when DSP block 120
20 is being used to perform its specialized function(s), the LFSR and the MISR capabilities of input registers 302 and output registers 304, respectively, may be disabled (or bypassed). During a validation phase, which may occur post fabrication at the factory or may occur after the
25 user has programed the device (i.e., before actual user operation or while the normal user operation has been temporarily suspended), input registers 302 can be configured as an input stimulus generator (e.g., by configuring input registers 302 in an LFSR mode) to
30 generate a test pattern that is fed as an input to DSP internal circuitry 300, whereas output registers 304 can be configured as an output data compressor (e.g., by

configuring output registers 304 in an MISR mode) to receive output test data from core circuitry 300 and to compress the received data into a unique signature that can be used to determine whether the DSP block 120 has
5 passed testing.

Input registers 302 with selectively enabled LFSR capability and output registers 304 with selectively enabled MISR capability are not built using soft logic but is instead built using specifically hard-wired non-
10 reconfigurable circuits. Configured in this way, the performance of the embedded BIST circuitry (e.g., input registers 302 and output registers 304) can be readily optimized and guaranteed without requiring lengthy time analysis operations, thereby dramatically simplifying the
15 integrated circuit design flow.

FIG. 3B is a diagram of circuit 302 that includes DSP input registers and that can be selectively enabled as a linear feedback shift register (LFSR) circuit. As shown in FIG. 3B, circuit 302 may include
20 input registers 310 (e.g., clock-triggered latches 310-1, 310-2, 310-3, 310-4, . . . , and 310-N), a multiplexer 312 associated with each input register 310, and a logic gate such as logic XOR (exclusive-OR) gate 314.

In the example of FIG. 3B, first input register
25 310-1 may be fed using first multiplexer 312, second input register 310-2 may be fed using second multiplexer 312, third input register 310-3 may be fed using third multiplexer 312, . . . , and Nth input registers 310-N may be fed using Nth multiplexer 312. First multiplexer 312
30 may have a first (0) input that receives external input signal In1 and a second (1) input that is coupled to the output of logic XOR gate 314. Similarly, second

multiplexer 312 may have a first (0) input that receives external input signal In2 and a second (1) input that is coupled to the output of the first input register 310-1. Third multiplexer 312 may have a first (0) input that
5 receives external input signal In3 and a second (1) input that is coupled to the output of the second input register 310-2. Input registers 310-1 may have corresponding outputs In1', In2', In3', . . . , and InN', respectively, that are fed to DSP internal circuitry 300 (FIG. 3A).
10 Input registers 310 connected in this way are sometimes considered to be connected in series, in a chain, or in a cascaded arrangement.

Each multiplexer 312 may have a control input that receives enable signal En_LFSR. Logic XOR gate 314
15 may have inputs that are connected to at least some of the outputs of the input registers. In the example of FIG. 3B, gate 314 has an input that is connected to the output of second input register 310-2 via path 316 and another input that is connected to the output of the last (most
20 significant bit) input register 310-N via path 318. These feedback connections 316 and 318 are sometimes referred to as "taps." For example, logic XOR gate 314 may be coupled to at least two taps, at least four taps, or other
suitable number of tap points (as indicated by dots 320).
25 In general, the taps may be connected to any subset of input register outputs.

Arranged in this way, circuit 302 may be configured to operate as a linear feedback shift register whenever enable signal En_LFSR is asserted (e.g., when
30 signal En_LFSR is driven high). During normal (non-testing) operations, enable signal En_LFSR may be deasserted (e.g., signal En_LFSR may be driven low) to

bypass the LFSR loop so that input registers 310 are used to simply latch incoming parallel data bits. The LFSR loop performance should be higher than performance of the DSP internal circuitry to ensure a sufficient hold time margin.

FIG. 3C is a diagram of circuit 304 that includes DSP output registers and that can be selectively enabled as a multiple input signature register (MISR) circuit. As shown in FIG. 3C, circuit 302 may include output registers 330 (e.g., clock-triggered latches 330-1, 330-2, 330-3, 330-4, . . . , and 330-M), a multiplexer 334 associated with each output register 330, and also a logic gate such as logic XOR gate 332 associated with each output register 330.

In the example of FIG. 3C, first output register 330-1 may be fed using first XOR gate 332-1, second output register 330-2 may be fed using second XOR 332-2, third output register 330-3 may be fed using third XOR gate 332-3, . . . , and Mth output registers 330-M may be fed using Mth XOR gate 332-M. First gate 332-1 may have a first input that receives output signal Out1' from the DSP internal circuitry and a second input that is coupled to the output of a first corresponding multiplexer 334. Similarly, second gate 332-2 may have a first input that receives output signal Out2' from the DSP internal circuitry and a second input that is coupled to the output of a second corresponding multiplexer 334. Third gate 332-3 may have a first input that receives output signal Out3' from the DSP internal circuitry and second input that is coupled to the output of a third corresponding multiplexer 334. Output registers 330-1 may have corresponding outputs Out1, Out2, Out3, . . . , and OutM,

respectively, that serve as outputs for DSP block 120 (FIG. 3A).

Each multiplexer 334 may have a first (0) input that is connected to a ground line (e.g., the first input may receive a logic "0" value), a second (1) input that is connected to the output of a preceding output registers, and a control input that receives a MISR enable signal En_MISR. In particular, the first multiplexer 334 associated with registers 330-1 has its second input connected to the output of the MSB (most significant bit) register 330-M. Output registers 330 connected in this way are sometimes considered to be connected in series, in a chain, or in a cascaded arrangement. The number of output registers may be different than the number of input registers (i.e., M may be different than N). If desired, the number of input registers may also be equal to the number of output registers.

Arranged in this way, circuit 304 may be configured to operate as a multiple input signature register whenever enable signal En_MISR is asserted (e.g., when signal En_MISR is driven high). During normal (non-testing) operations, enable signal En_MISR may be deasserted (e.g., signal En_LFSR may be driven low) to bypass the MISR loop so that output registers 330 are used to simply latch outgoing parallel data bits. The MISR performance should be higher than performance of the DSP internal circuitry to ensure a sufficient hold time margin.

Computer-aided design (CAD) tools in a circuit design system may be used to design a programmable integrated circuit such as a programmable logic device that includes DSP blocks with embedded BIST circuitry.

An illustrative circuit design system 400 in accordance with an embodiment is shown in FIG. 4. Circuit design system 400 may be implemented on integrated circuit design computing equipment. For example, system
5 400 may be based on one or more processors such as personal computers, workstations, etc. The processor(s) may be linked using a network (e.g., a local or wide area network). Memory in these computers or external memory and storage devices such as internal and/or external hard
10 disks may be used to store instructions and data.

Software-based components such as computer-aided design tools 420 and databases 430 reside on system 400. During operation, executable software such as the software of computer aided design tools 420 runs on the
15 processor(s) of system 400. Databases 430 are used to store data for the operation of system 400. In general, software and data may be stored on any computer-readable medium (storage) in system 400. Such storage may include computer memory chips, removable and fixed media such as
20 hard disk drives, flash memory, compact discs (CDs), digital versatile discs (DVDs), Blu-ray discs (BDs), other optical media, and floppy diskettes, tapes, or any other suitable memory or storage device(s). When the software of system 400 is installed, the storage of system 400 has
25 instructions and data that cause the computing equipment in system 400 to execute various methods (processes). When performing these processes, the computing equipment is configured to implement the functions of the circuit design system.

30 The computer aided design (CAD) tools 420, some or all of which are sometimes referred to collectively as a CAD tool, a circuit design tool, or an electronic design

automation (EDA) tool, may be provided by a single vendor or by multiple vendors. Tools 420 may be provided as one or more suites of tools (e.g., a compiler suite for performing tasks associated with implementing a circuit design in a programmable logic device) and/or as one or more separate software components (tools). Database(s) 430 may include one or more databases that are accessed only by a particular tool or tools and may include one or more shared databases. Shared databases may be accessed by multiple tools. For example, a first tool may store data for a second tool in a shared database. The second tool may access the shared database to retrieve the data stored by the first tool. This allows one tool to pass information to another tool. Tools may also pass information between each other without storing information in a shared database if desired.

Illustrative computer aided design tools 520 that may be used in a circuit design system such as circuit design system 400 of FIG. 4 are shown in FIG. 5.

The design process may start with the formulation of functional specifications of the integrated circuit design (e.g., a functional or behavioral description of the integrated circuit design). A circuit designer may specify the functional operation of a desired circuit design using design and constraint entry tools 564. Design and constraint entry tools 564 may include tools such as design and constraint entry aid 566 and design editor 568. Design and constraint entry aids such as aid 566 may be used to help a circuit designer locate a desired design from a library of existing circuit designs and may provide computer-aided assistance to the circuit

designer for entering (specifying) the desired circuit design.

As an example, design and constraint entry aid 566 may be used to present screens of options for a user. The user may click on on-screen options to select whether the circuit being designed should have certain features. Design editor 568 may be used to enter a design (e.g., by entering lines of hardware description language code), may be used to edit a design obtained from a library (e.g., using a design and constraint entry aid), or may assist a user in selecting and editing appropriate prepackaged code/designs.

Design and constraint entry tools 564 may be used to allow a circuit designer to provide a desired circuit design using any suitable format. For example, design and constraint entry tools 564 may include tools that allow the circuit designer to enter a circuit design using truth tables. Truth tables may be specified using text files or timing diagrams and may be imported from a library. Truth table circuit design and constraint entry may be used for a portion of a large circuit or for an entire circuit.

As another example, design and constraint entry tools 564 may include a schematic capture tool. A schematic capture tool may allow the circuit designer to visually construct integrated circuit designs from constituent parts such as logic gates and groups of logic gates. Libraries of preexisting integrated circuit designs may be used to allow a desired portion of a design to be imported with the schematic capture tools.

If desired, design and constraint entry tools 564 may allow the circuit designer to provide a circuit

design to the circuit design system 400 using a hardware description language such as Verilog hardware description language (Verilog HDL), Very High Speed Integrated Circuit Hardware Description Language (VHDL), SystemVerilog, or a
5 higher-level circuit description language such as OpenCL or SystemC, just to name a few. The designer of the integrated circuit design can enter the circuit design by writing hardware description language code with editor 568. Blocks of code may be imported from user-maintained
10 or commercial libraries if desired.

After the design has been entered using design and constraint entry tools 564, behavioral simulation tools 572 may be used to simulate the functional performance of the circuit design. If the functional
15 performance of the design is incomplete or incorrect, the circuit designer can make changes to the circuit design using design and constraint entry tools 564. The functional operation of the new circuit design may be verified using behavioral simulation tools 572 before
20 synthesis operations have been performed using tools 574. Simulation tools such as behavioral simulation tools 572 may also be used at other stages in the design flow if desired (e.g., after logic synthesis). The output of the behavioral simulation tools 572 may be provided to the
25 circuit designer in any suitable format (e.g., truth tables, timing diagrams, etc.).

Once the functional operation of the circuit design has been determined to be satisfactory, logic synthesis and optimization tools 574 may generate a gate-
30 level netlist of the circuit design, for example using gates from a particular library pertaining to a targeted process supported by a foundry, which has been selected to

produce the integrated circuit. Alternatively, logic synthesis and optimization tools 574 may generate a gate-level netlist of the circuit design using gates of a targeted programmable logic device (i.e., in the logic and interconnect resources of a particular programmable logic device product or product family).

Logic synthesis and optimization tools 574 may optimize the design by making appropriate selections of hardware to implement different logic functions in the circuit design based on the circuit design data and constraint data entered by the logic designer using tools 564. As an example, logic synthesis and optimization tools 574 may perform register retiming on the circuit design based on the length of a combinational path between registers in the circuit design and corresponding timing constraints that were entered by the logic designer using tools 564.

After logic synthesis and optimization using tools 574, the circuit design system may use tools such as placement and routing tools 576 to perform physical design steps (layout synthesis operations). Placement and routing tools 576 are used to determine where to place each gate of the gate-level netlist produced by tools 574. For example, if two counters interact with each other, the placement and routing tools 576 may locate these counters in adjacent regions to reduce interconnect delays or to satisfy timing requirements specifying the maximum permitted interconnect delay. The placement and routing tools 576 create orderly and efficient implementations of circuit designs for any targeted integrated circuit (e.g., for a given programmable integrated circuit such as a field-programmable gate array (FPGA)).

Tools such as tools 574 and 576 may be part of a compiler suite (e.g., part of a suite of compiler tools provided by a programmable logic device vendor). In certain embodiments, tools such as tools 574, 576, and 578
5 may also include timing analysis tools such as timing estimators. This allows tools 574 and 576 to satisfy performance requirements (e.g., timing requirements) before actually producing the integrated circuit.

After an implementation of the desired circuit
10 design has been generated using placement and routing tools 576, the implementation of the design may be analyzed and tested using analysis tools 578. For example, analysis tools 578 may include timing analysis tools, power analysis tools, or formal verification tools,
15 just to name few.

After satisfactory optimization operations have been completed using tools 520 and depending on the targeted integrated circuit technology, tools 520 may produce a mask-level layout description of the integrated
20 circuit or configuration data for programming the programmable logic device.

Illustrative operations involved in using tools 520 of FIG. 5 to produce the mask-level layout description of the integrated circuit are shown in FIG. 6. As shown
25 in FIG. 6, a circuit designer may first provide a design specification 602. The design specification 602 may, in general, be a behavioral description provided in the form of an application code (e.g., C code, C++ code, SystemC code, OpenCL code, etc.). In some scenarios, the design
30 specification may be provided in the form of a register transfer level (RTL) description 606.

The RTL description may have any form of describing circuit functions at the register transfer level. For example, the RTL description may be provided using a hardware description language such as the Verilog hardware description language (Verilog HDL or Verilog),
5 the SystemVerilog hardware description language (SystemVerilog HDL or SystemVerilog), or the Very High Speed Integrated Circuit Hardware Description Language (VHDL). If desired, a portion or all of the RTL
10 description may be provided as a schematic representation.

In general, the behavioral design specification 602 may include untimed or partially timed functional code (i.e., the application code does not describe cycle-by-cycle hardware behavior), whereas the RTL description 606
15 may include a fully timed design description that details the cycle-by-cycle behavior of the circuit at the register transfer level.

Design specification 602 or RTL description 606 may also include target criteria such as area use, power
20 consumption, delay minimization, clock frequency optimization, or any combination thereof. The optimization constraints and target criteria may be collectively referred to as constraints.

Those constraints can be provided for individual
25 data paths, portions of individual data paths, portions of a design, or for the entire design. For example, the constraints may be provided with the design specification 602, the RTL description 606 (e.g., as a pragma or as an assertion), in a constraint file, or through user input
30 (e.g., using the design and constraint entry tools 564 of FIG. 5), to name a few.

At step 604, behavioral synthesis (sometimes also referred to as algorithmic synthesis) may be performed to convert the behavioral description into an RTL description 606. Step 604 may be skipped if the
5 design specification is already provided in form of an RTL description.

At step 618, behavioral simulation tools 572 may perform an RTL simulation of the RTL description, which may verify the functional performance of the RTL
10 description. If the functional performance of the RTL description is incomplete or incorrect, the circuit designer can make changes to the HDL code (as an example). During RTL simulation 618, actual results obtained from simulating the behavior of the RTL description may be
15 compared with expected results.

During step 608, logic synthesis operations may generate gate-level description 610 using logic synthesis and optimization tools 574 from FIG. 5. If desired, logic synthesis operations may perform register retiming as
20 illustrated in FIG. 2 according to the constraints that are included in design specification 602 or RTL description 606.

During step 612, physical synthesis operations (e.g., place and route and optimization operations using
25 for example placement and routing tools 576) may place and connect the different gates in gate-level description 610 in a preferred location on the targeted integrated circuit to meet given target criteria (e.g., minimize area and maximize routing efficiency or minimize path delay and
30 maximize clock frequency or any combination thereof).

If desired, physical synthesis operations may perform register retiming as illustrated in FIG. 2 and

according to the constraints that are included in design specification 602 or RTL description 606. As an example, physical synthesis operations may perform register retiming by changing the configuration of some pipelined routing resources (e.g., some instances of pipelined routing resource 300 of FIG. 3) from operating in pipeline register mode to operating in non-pipelined mode and the configuration of other pipelined routing resources (e.g., other instances of pipelined routing resources 300 of FIG. 3) from operating in non-pipelined mode to operating in pipeline register mode. The output of physical synthesis 612 is a mask-level layout description 616.

The overall integrated circuit design flow described in connection with FIG. 6 can be dramatically simplified since the embedded BIST circuitry is able to easily meeting the timing requirements associated with the DSP block, which substantially reduces the complexity of the timing analysis step 614 in FIG. 6. In practice, the performance of all DSP blocks on an integrated circuit can be concurrently characterized using the embedded BIST circuitry.

FIG. 7 is a flow chart of illustrative steps for performing validation on a specialized processing (DSP) block with embedded built-in self-test circuitry in accordance with an embodiment. At step 700, a configuration bit stream may be loaded into a programmable integrated circuit die, configuring the IC die to perform a custom user function.

At step 702, a seed may be injected into the DSP input registers through the DSP input signal path by clocking the input registers once while signal En_LFSR is deasserted. For example, an initializing seed value of

"101011 . . . " may be presented as parallel input bits to In1, In2, In3, . . . , InN, and these values may be latched by input registers 310.

At step 704, both control signals En_LFSR and En_MISR may be simultaneously asserted to enable the LFSR and the MISR built-in self-test functionality of the input and output registers.

At step 706, the DSP block may be allowed to run for X clock cycles (e.g., for a million clock cycles, a thousand clock cycles, for ten clock cycles, for one clock cycle, etc.) at a selected frequency. In the first iteration, the selected frequency may be relatively low to ensure that the DSP block performs as expected. During this time, circuit 302 configured as an LFSR will change its combinational state while circuit 304 configured as an MISR will output a new compressed signature every clock cycle. Circuits 302 and 304 and core DSP circuitry 300 are controlled by the same clock and therefore run at the same frequency. In general, more than one DSP blocks can be initialized and tested in parallel (e.g., the BIST circuitry in multiple DSP blocks may be enabled concurrently).

After X clock cycles has passed, the final compressed MISR output can be compared to a predetermined signature associated with the original seed value (step 708). The predetermined signature may represent a theoretical value that is pre-computed based on a properly functioning DSP block. If the signatures match, the operating frequency of the DSP block may be increased (step 710), and processing may loop back to step 702 for the next iteration (as indicated by path 712).

If the signatures no longer match, the maximum

operating frequency (F_{max}) of the DSP block will have been reached, and the BIST operation for that DSP block is complete (step 714). If desired, an F_{max} value determined in this way can be correlated with other performance
5 metrics such as the ring oscillator frequency on an integrated circuit and can therefore sometimes be used to bin the performance of integrated circuit dies.

The steps of FIG. 7 are merely illustrative. The existing steps may be modified or omitted; some of the
10 steps may be performed in parallel; additional steps may be added; and the order of certain steps may be reversed or altered.

The embodiments thus far have been described with respect to integrated circuits. The methods and
15 apparatuses described herein may be incorporated into any suitable circuit. For example, they may be incorporated into numerous types of devices such as programmable logic devices, application specific standard products (ASSPs), and application specific integrated circuits (ASICs).
20 Examples of programmable logic devices include programmable arrays logic (PALs), programmable logic arrays (PLAs), field programmable logic arrays (FPLAs), electrically programmable logic devices (EPLDs), electrically erasable programmable logic devices (EEPLDs),
25 logic cell arrays (LCAs), complex programmable logic devices (CPLDs), and field programmable gate arrays (FPGAs), just to name a few.

The programmable logic device described in one or more embodiments herein may be part of a data
30 processing system that includes one or more of the following components: a processor; memory; IO circuitry; and peripheral devices. The data processing can be used

in a wide variety of applications, such as computer networking, data networking, instrumentation, video processing, digital signal processing, or any suitable other application where the advantage of using programmable or re-programmable logic is desirable. The programmable logic device can be used to perform a variety of different logic functions. For example, the programmable logic device can be configured as a processor or controller that works in cooperation with a system processor. The programmable logic device may also be used as an arbiter for arbitrating access to a shared resource in the data processing system. In yet another example, the programmable logic device can be configured as an interface between a processor and one of the other components in the system. In one embodiment, the programmable logic device may be one of the family of devices owned by ALTERA Corporation.

Example:

Example 1 is an integrated circuit die, comprising: memory circuitry; configurable logic circuitry; and a specialized processing block that includes embedded built-in self-test circuitry.

Example 2 is the integrated circuit die of example 1, wherein the specialized processing block optionally comprises a digital signal processor (DSP) block.

Example 3 is the integrated circuit die of example 1, wherein the embedded built-in self-test circuitry within the specialized processing block optionally includes input registers coupled in a loop.

Example 4 is the integrated circuit die of example 3, wherein the input registers of the specialized

processing block are optionally selectively enabled to operate as a linear feedback shift register.

Example 5 is the integrated circuit die of example 4, wherein the specialized processing block
5 optionally further comprises a plurality of multiplexers interposed in the loop.

Example 6 is the integrated circuit die of example 5, wherein each multiplexer in the plurality of multiplexers is optionally controlled by a common enable
10 signal.

Example 7 is the integrated circuit die of example 4, wherein the specialized processing block optionally further comprises a logic exclusive-OR gate interposed in the loop.

15 Example 8 is the integrated circuit die of example 1, wherein the embedded built-in self-test circuitry within the specialized processing block optionally includes output registers coupled in a chain.

Example 9 is the integrated circuit
20 die of example 8, wherein the output registers of the specialized processing block are optionally selectively enabled to operate as a multiple input signature register.

Example 10 is the integrated circuit die of example 9, wherein the specialized processing block
25 optionally further comprises a plurality of logic exclusive-OR gates interposed in the chain.

Example 11 is the integrated circuit die of example 10, wherein the specialized processing block optionally further comprises a plurality of
30 multiplexers feeding the plurality of logic exclusive-OR gates.

Example 12 is the integrated circuit die of example 11, wherein each multiplexer in the plurality of multiplexers has an input that optionally receives a ground power supply voltage.

5 Example 13 is a method of operating an integrated circuit that includes memory circuitry and a specialized processing block, the method comprising: with the memory circuitry, receiving a configuration bit stream; with input registers embedded within the
10 specialized processing block, performing built-in self-test (BIST) operations to validate the functionality of the specialized processing block; and with output registers embedded within the specialized processing block, performing the built-in self-test (BIST) operations
15 to validate the functionality of the specialized processing block.

Example 14 is the method of example 13, wherein performing the BIST operations with the input registers optionally comprises configuring the input registers in a
20 linear feedback shift register mode, and wherein performing the BIST operations with the output registers comprises configuring the output registers in a multiple input signature register mode.

Example 15 is the method of example 14,
25 optionally further comprising: configuring the input registers using a first enable signal; configuring the output registers using a second enable signal; feeding a seed BIST value into the input registers while the first enable signal is deasserted; and after latching the seed
30 BIST value with the input register, asserting both the first and second enable signals.

Example 16 is the method of example 15,
optionally further comprising: after asserting the first
and second enable signals, waiting for a predetermined
time period; and after waiting for the predetermined time
5 period, comparing a signature generated by the output
registers configured in the multiple input signature
register mode to a pre-computed signature.

Example 17 is the method of example 16,
optionally further comprising: in response to determining
10 that the signature generated by the output registers
matches the pre-computed signature, increasing the
frequency at which the specialized processing block is
operated; and in response to determining that the
signature generated by the output registers does not match
15 the pre-computed signature, terminating the BIST
operations.

Example 18 is an integrated circuit
die, comprising: volatile memory; soft logic; and a
digital signal processing block with embedded built-in
20 self-test circuitry, wherein the built-in self-test
circuitry is not implemented using any soft logic.

Example 19 is the integrated circuit die of
example 18, wherein the built-in self-test circuitry of
the digital signal processing block is optionally
25 implemented using input registers and output registers
within the digital signal processing block, and wherein
the digital signal processing block includes internal
processing circuitry that is coupled between the input
registers and the output registers.

30 Example 20 is the integrated circuit die of
example 19, wherein the input registers are selectively
enabled to operate as a linear feedback shift register,

and wherein the output registers are selectively enabled to operate as a multiple input signature register.

For instance, all optional features of the apparatus described above may also be implemented with
5 respect to the method or process described herein. The foregoing is merely illustrative of the principles of this invention and various modifications can be made by those skilled in the art. The foregoing embodiments may be implemented individually or in any combination.

10

What is Claimed is:

1. An integrated circuit die, comprising:
memory circuitry;
configurable logic circuitry; and
a specialized processing block that
includes embedded built-in self-test circuitry.
2. The integrated circuit die of claim 1,
wherein the specialized processing block comprises a
digital signal processor (DSP) block.
3. The integrated circuit die of claim 1,
wherein the embedded built-in self-test circuitry within
the specialized processing block includes input registers
coupled in a loop.
4. The integrated circuit die of claim 3,
wherein the input registers of the specialized processing
block are selectively enabled to operate as a linear
feedback shift register.
5. The integrated circuit die of claim 4,
wherein the specialized processing block further
comprises:
a plurality of multiplexers interposed in
the loop.
6. The integrated circuit die of claim 5,
wherein each multiplexer in the plurality of multiplexers
is controlled by a common enable signal.

7. The integrated circuit die of claim 4, wherein the specialized processing block further comprises:

a logic exclusive-OR gate interposed in the loop.

8. The integrated circuit die of claim 1, wherein the embedded built-in self-test circuitry within the specialized processing block includes output registers coupled in a chain.

9. The integrated circuit die of claim 8, wherein the output registers of the specialized processing block are selectively enabled to operate as a multiple input signature register.

10. The integrated circuit die of claim 9, wherein the specialized processing block further comprises:

a plurality of logic exclusive-OR gates interposed in the chain.

11. The integrated circuit die of claim 10, wherein the specialized processing block further comprises:

a plurality of multiplexers feeding the plurality of logic exclusive-OR gates.

12. The integrated circuit die of claim 11, wherein each multiplexer in the plurality of multiplexers has an input that receives a ground power supply voltage.

13. A method of operating an integrated circuit that includes memory circuitry and a specialized processing block, the method comprising:

with the memory circuitry, receiving a configuration bit stream;

with input registers embedded within the specialized processing block, performing built-in self-test (BIST) operations to validate the functionality of the specialized processing block; and

with output registers embedded within the specialized processing block, performing the built-in self-test (BIST) operations to validate the functionality of the specialized processing block.

14. The method of claim 13, wherein performing the BIST operations with the input registers comprises configuring the input registers in a linear feedback shift register mode, and wherein performing the BIST operations with the output registers comprises configuring the output registers in a multiple input signature register mode.

15. The method of claim 14, further comprising:
configuring the input registers using a first enable signal;

configuring the output registers using a second enable signal;

feeding a seed BIST value into the input registers while the first enable signal is deasserted; and

after latching the seed BIST value with the input register, asserting both the first and second enable signals.

16. The method of claim 15, further comprising:
after asserting the first and second enable signals, waiting for a predetermined time period; and
after waiting for the predetermined time period, comparing a signature generated by the output registers configured in the multiple input signature register mode to a pre-computed signature.

17. The method of claim 16, further comprising:

in response to determining that the signature generated by the output registers matches the pre-computed signature, increasing the frequency at which the specialized processing block is operated; and

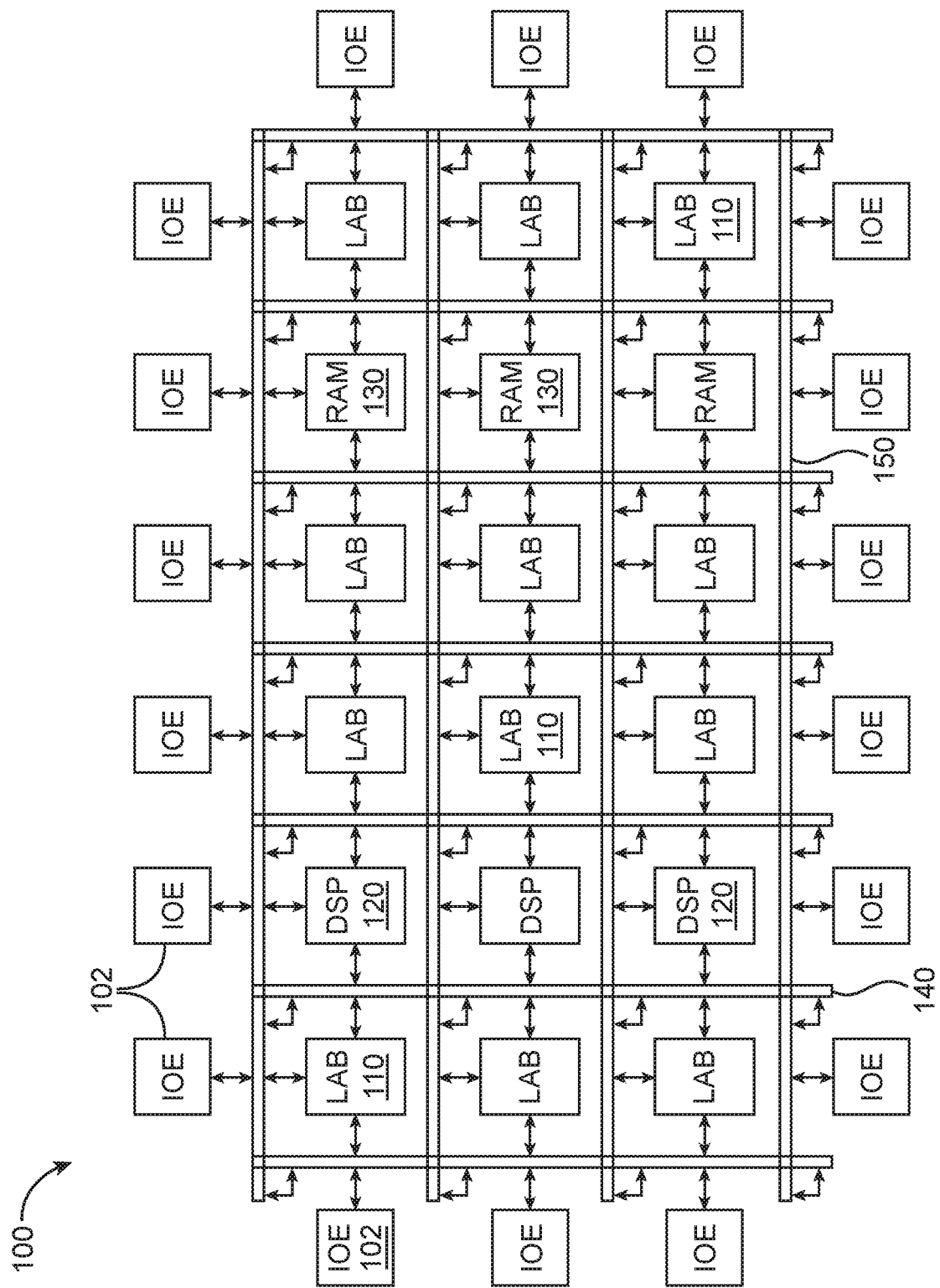
in response to determining that the signature generated by the output registers does not match the pre-computed signature, terminating the BIST operations.

18. An integrated circuit die, comprising:
volatile memory;
soft logic; and
a digital signal processing block with embedded built-in self-test circuitry, wherein the built-in self-test circuitry is not implemented using any soft logic.

19. The integrated circuit die of claim 18, wherein the built-in self-test circuitry of the digital signal processing block is implemented using input registers and output registers within the digital signal processing block, and wherein the digital signal

processing block includes internal processing circuitry that is coupled between the input registers and the output registers.

20. The integrated circuit die of claim 19, wherein the input registers are selectively enabled to operate as a linear feedback shift register, and wherein the output registers are selectively enabled to operate as a multiple input signature register.



2 / 9

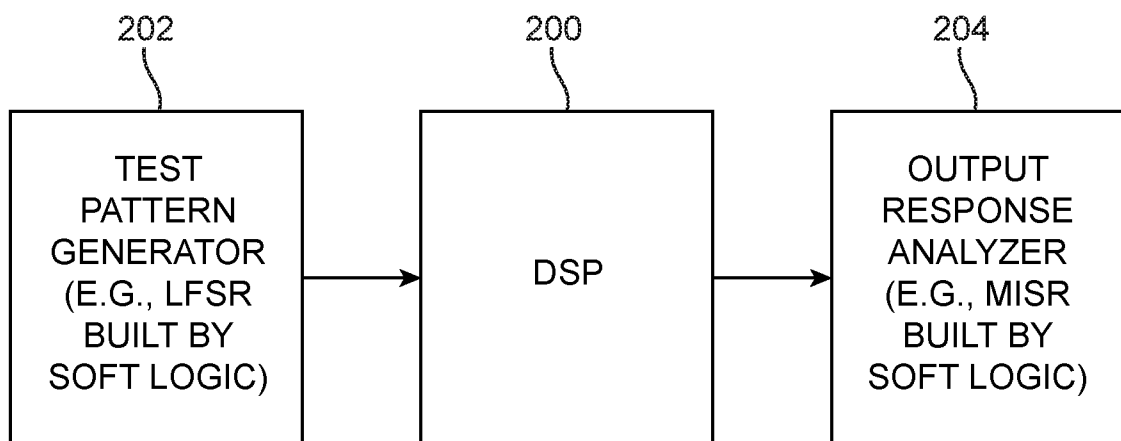


FIG. 2

3 / 9

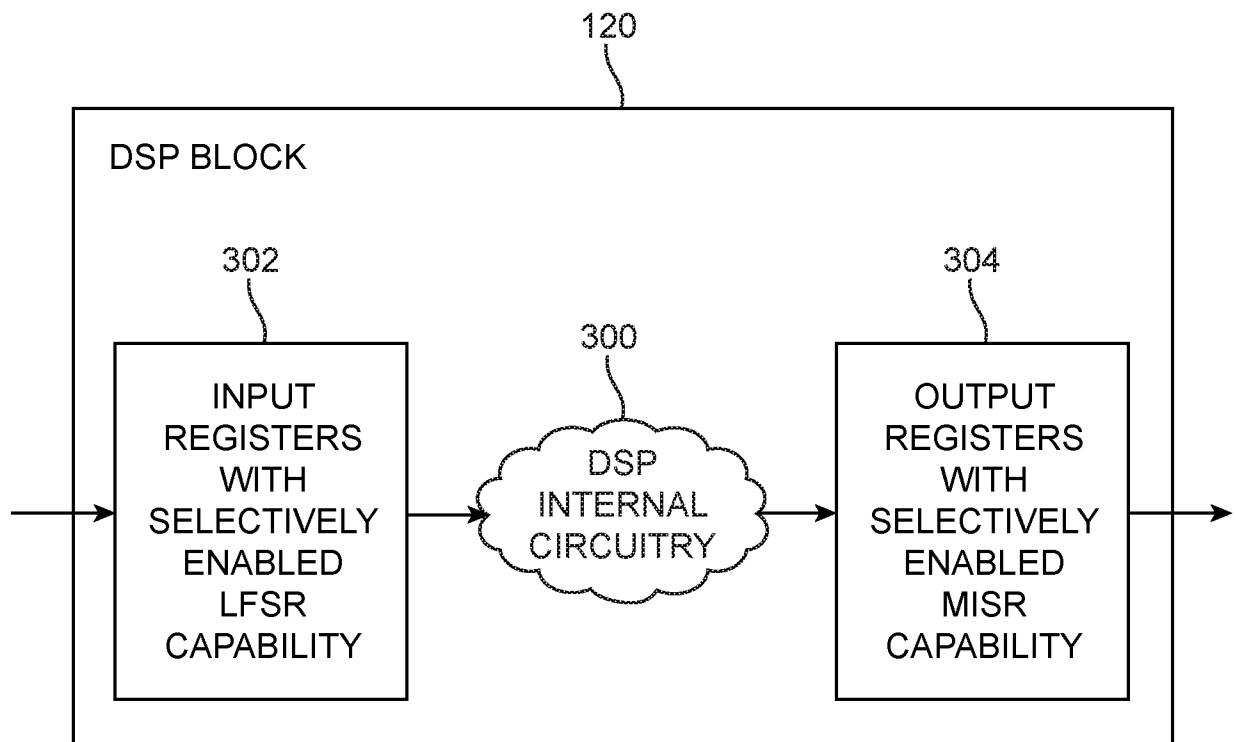


FIG. 3A

4 / 9

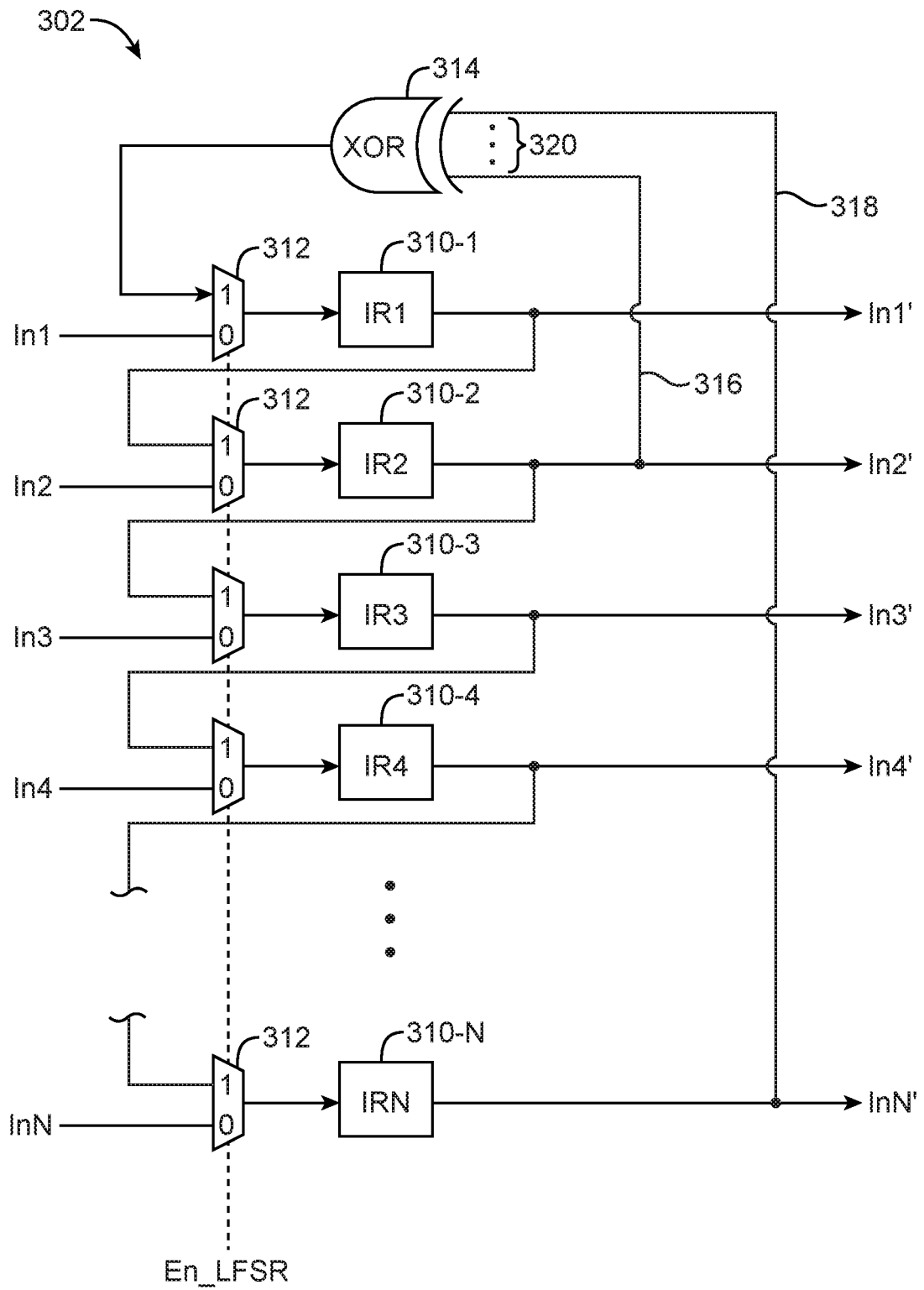
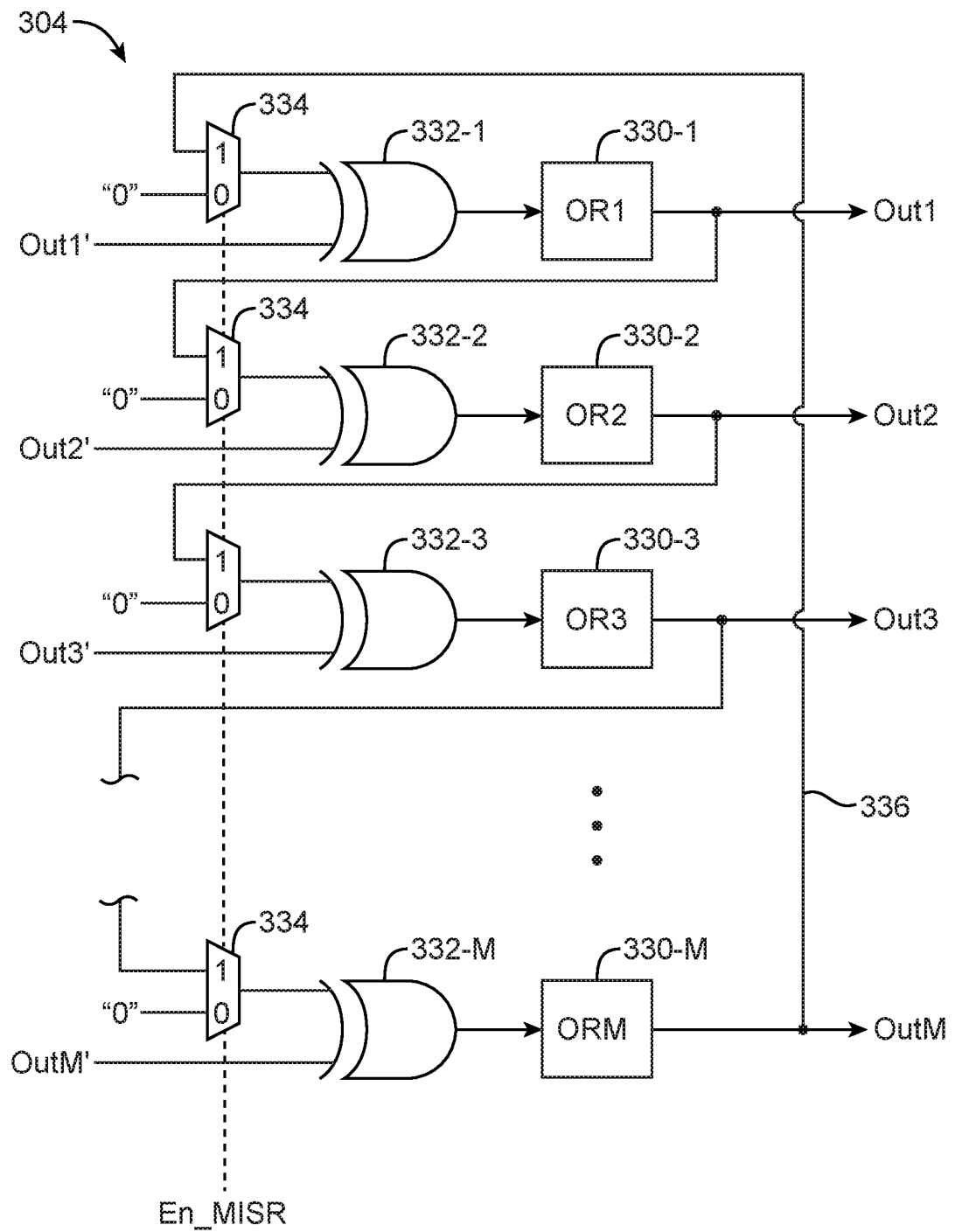


FIG. 3B

5 / 9



6 / 9

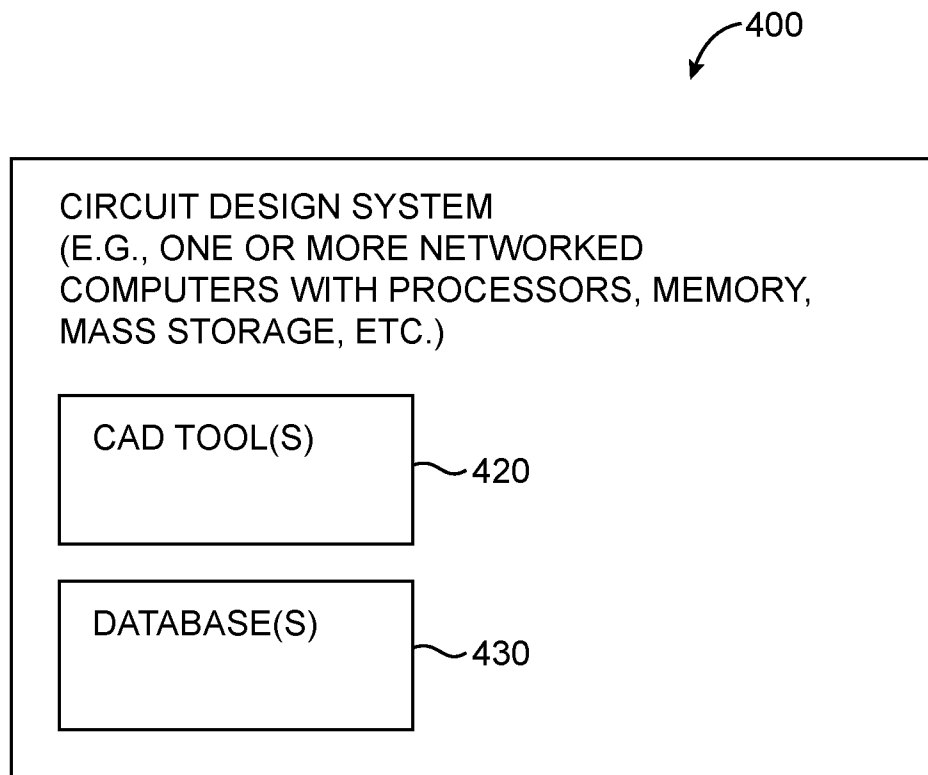


FIG. 4

7 / 9

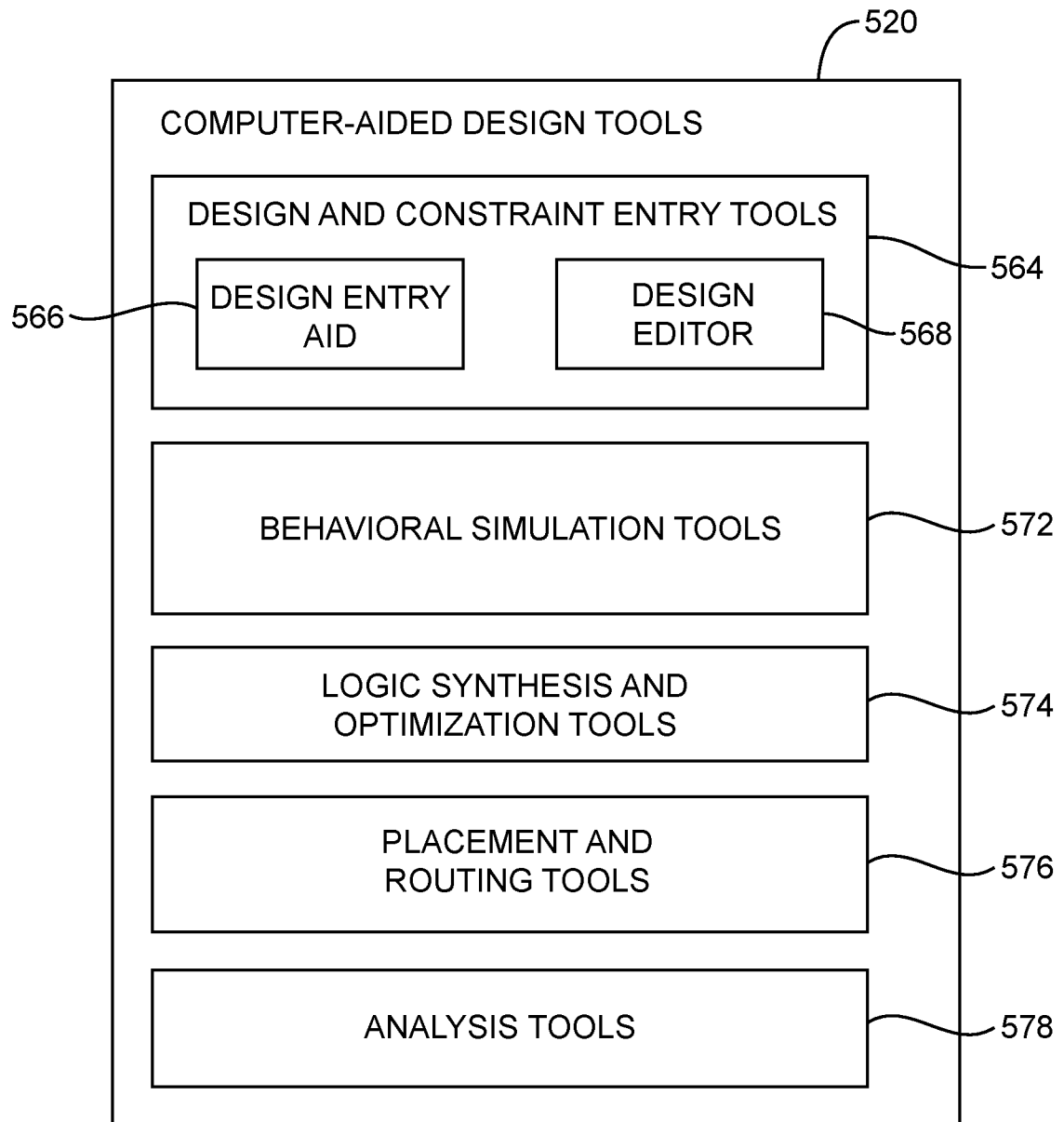


FIG. 5

8 / 9

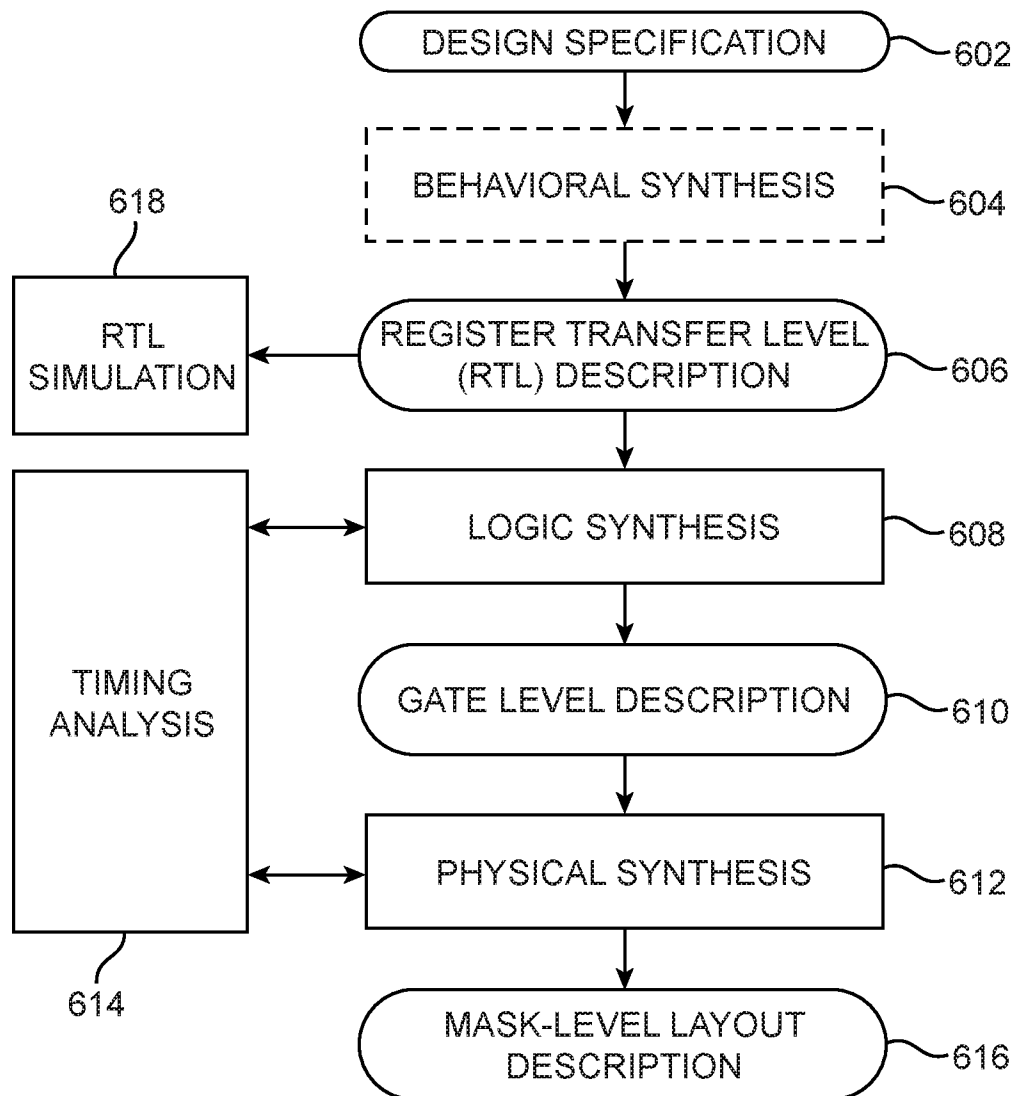


FIG. 6

9 / 9

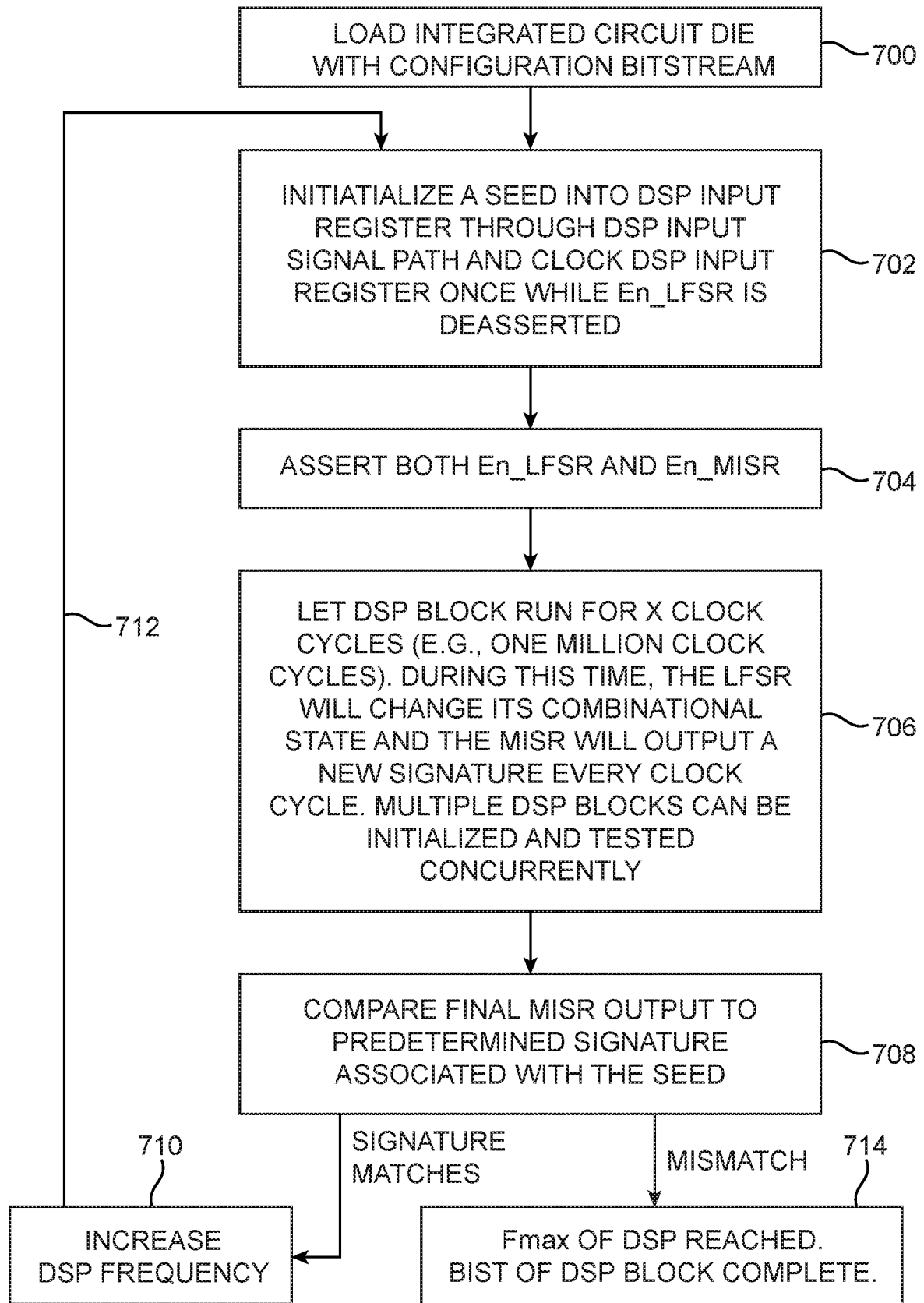


FIG. 7

A. CLASSIFICATION OF SUBJECT MATTER**G01R 31/317(2006.01)i, G01R 31/3185(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; G01R 31/26; G01R 31/28; G06F 17/00; G06F 11/00; G01R 31/3185

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: integrated, circuit, memory, logic, BIST, built-in, self-test

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 8549368 B1 (DEOGHARIA et al.) 01 October 2013 See column 1, line 23-column 8, line 14, claims 1-3 and figures 1,2.	1-3,8-19
Y		4-7,20
Y	US 2011-0175638 A1 (MAEDA) 21 July 2011 See paragraph [30], claims 9,10 and figure 1.	4-7,20
A	WO 00-11566 A1 (CREDENCE SYSTEMS CORPORATION) 02 March 2000 See claims 1-20 and figure 1A.	1-20
A	US 8499208 B2 (DIEFFENDERFER et al.) 30 July 2013 See claims 1-20 and figures 1-8.	1-20
A	US 6564349 B1 (MITTEN et al.) 13 May 2003 See claims 1-25 and figures 1-4.	1-20



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

27 June 2017 (27.06.2017)

Date of mailing of the international search report

28 June 2017 (28.06.2017)

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/US2017/027238

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 8549368 B1	01/10/2013	None	
US 2011-0175638 A1	21/07/2011	JP 2011-149775 A	04/08/2011
WO 00-11566 A1	02/03/2000	DE 69826727 T2	06/10/2005
		EP 1129415 A1	05/09/2001
		EP 1129415 A4	06/08/2003
		EP 1129415 B1	29/09/2004
		JP 2002-523894 A	30/07/2002
		KR 10-0538286 B1	21/12/2005
US 8499208 B2	30/07/2013	US 2008-115026 A1	15/05/2008
US 6564349 B1	13/05/2003	AU 2001-30985 A1	03/09/2001
		WO 01-63827 A1	30/08/2001