



(51) International Patent Classification:

G11C 8/00 (2006.01) *H03L 7/081* (2006.01)
G11C 7/22 (2006.01)

(21) International Application Number:

PCT/US2011/038361

(22) International Filing Date:

27 May 2011 (27.05.2011)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

12/793,533 3 June 2010 (03.06.2010) US

(71) Applicant (for all designated States except US): **INTEL CORPORATION** [US/US]; 2200 Mission College Boulevard, MS: RNB-4-150, Santa Clara, California 95052 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **MOSALIKANTI, Praveen** [IN/US]; 11322 Nw Kimble Court, Portland, Oregon 97229 (US). **KURD, Nasser A.** [US/US]; 4284 NW 125th Avenue, Portland, Oregon 97229 (US). **MOZAK, Christopher P.** [US/US]; 16648 SW Delta Court, Beaverton, Oregon 97006 (US).

(74) Agents: **VINCENT, Lester J.** et al.; Blakely Sokoloff Taylor & Zafman, 1279 Oakmead Parkway, Sunnyvale, California 94085 (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, 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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, 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, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, 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, ML, MR, NE, SN, TD, TG).

Published:

— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: METHODS AND APPARATUSES FOR DELAY-LOCKED LOOPS AND PHASE-LOCKED LOOPS

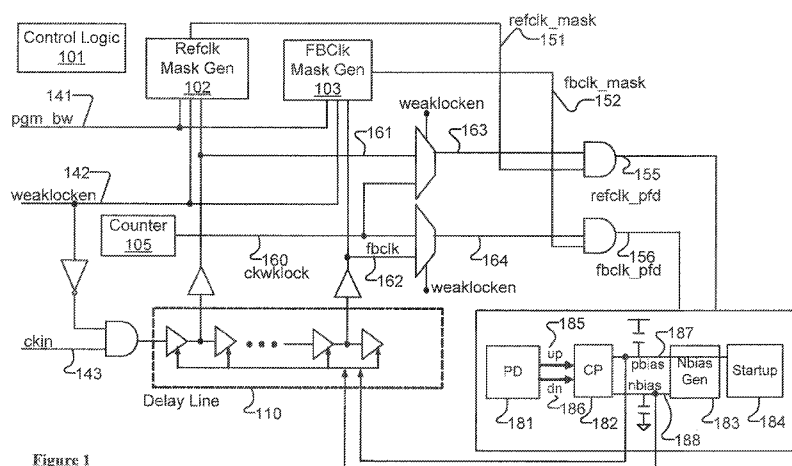


Figure 1

(57) Abstract: A low power delay-locked loop (DLL) is presented. In one embodiment, the DLL includes a phase detector which includes a reference input and a feedback input to determine a phase difference. The DLL also includes a controller to determine whether to provide a signal to both the reference input and the feedback input such that the reference input and the feedback input receive an identical input, for example, during low power operation.

METHODS AND APPARATUSES FOR DELAY-LOCKED LOOPS AND PHASE-LOCKED LOOPS

FIELD OF THE INVENTION

5 Embodiments of the invention relate to a delay-locked loop or a phase-locked loop; more particularly, a low power delay-locked loop or a low power phase-locked loop.

BACKGROUND OF THE INVENTION

 DDR interfaces typically use several DLLs (DLLs) where each DLL drives several phase interpolators (PIs). Each PI is used to provide adjustable delay on a pin or a set of
10 pins to help meet timing constraints.

 During a period of inactivity, the DLLs are not shut down because of the long relock time involved (typically on the order of 100-200 ns). Some circuitry only powers down the DLL if the circuitry is shut down for long enough to allow the DLL to relock after being powered up.

15 In power down modes (e.g., DDR CKE power down), the exit latency frequently cannot be more than 10-20 ns. DLLs are not likely to be powered down if such a requirement exists. The total current consumed by the all the DLLs in a DDR port may be about 100 mA. This leads to a waste of power usage when the circuitry is inactive.

 Often, DLLs can be reused in more than one application but with different
20 bandwidth requirements. A DLL with a programmable bandwidth is suitable for design reuse which would greatly save time and cost of development.

BRIEF DESCRIPTION OF THE DRAWINGS

 Embodiments of the present invention will be understood more fully from the
25 detailed description given below and from the accompanying drawings of various embodiments of the invention, which, however, should not be taken to limit the invention to the specific embodiments, but are for explanation and understanding only.

Figure 1 is a block diagram of a low power delay-locked loop in accordance with one embodiment of the invention.

30 **Figure 2** is a waveform diagram for a weaklock operation in accordance with one embodiment of the invention.

Figure 3 is a waveform diagram for a programmable frequency operation in accordance with one embodiment of the invention.

Figure 4 is a flow diagram of one embodiment of a process for a low power delay-locked loop.

Figure 5 illustrates a computer system for use with one embodiment of the present invention.

5

DETAILED DESCRIPTION OF THE INVENTION

Methods and apparatuses for delay-locked loops (DLLs) and phase-locked loops (PLLs) are presented with the reference to a delay-locked loop (DLL). In one embodiment, the DLL includes a phase detector which includes a reference input and a feedback input to determine a phase difference. The DLL also includes a controller to determine whether to provide a signal to both the reference input and the feedback input such that the reference input and the feedback input receive an identical input, for example, during low power operation.

In the following description, numerous details are set forth to provide a more thorough explanation of embodiments of the present invention. It will be apparent, however, to one skilled in the art, that embodiments of the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form, rather than in detail, in order to avoid obscuring embodiments of the present invention.

Some portions of the detailed descriptions which follow are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the

following discussion, it is appreciated that throughout the description, discussions utilizing terms such as "processing" or "computing" or "calculating" or "determining" or "displaying" or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

Embodiments of present invention also relate to apparatuses for performing the operations herein. Some apparatuses may be specially constructed for the required purposes, or it may comprise a general purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a computer readable storage medium, such as, but not limited to, any type of disk including floppy disks, optical disks, CD-ROMs, DVD-ROMs, and magnetic-optical disks, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, NVRAMs, magnetic or optical cards, or any type of media suitable for storing electronic instructions, and each coupled to a computer system bus.

The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various general purpose systems may be used with programs in accordance with the teachings herein, or it may prove convenient to construct more specialized apparatus to perform the required method steps. The required structure for a variety of these systems will appear from the description below. In addition, embodiments of the present invention are not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the invention as described herein.

A machine-readable medium includes any mechanism for storing or transmitting information in a form readable by a machine (e.g., a computer). For example, a machine-readable medium includes read only memory ("ROM"); random access memory ("RAM"); magnetic disk storage media; optical storage media; flash memory devices; etc.

The method and the apparatus described herein are primarily discussed in reference to low power DLLs. However, the method and the apparatus are not limited to low power DLLs, they may be implemented in conjunction with low power PLLs. In addition, lower power DLLs are primarily discussed in reference to a processor computer system. They may be implemented on or in association with any integrated circuit device or system,

such as cell phones, personal digital assistants, embedded controllers, mobile platforms, desktop platforms, and server platforms.

Overview

5 Methods and apparatuses for DLLs and PLLs are presented with the reference to a low power delay-locked loop (DLL). In one embodiment, the DLL includes a phase detector which includes a reference input and a feedback input to determine a phase difference. The DLL also includes a controller to determine whether to provide a signal to both the reference input and the feedback input such that the reference input and the
10 feedback input receive an identical input, for example, during low power operation.

 Figure 1 is a block diagram of a low power delay-locked loop (DLL) in accordance with one embodiment of the invention. Many related components such as buses and peripherals have not been shown to avoid obscuring the invention. Referring to Figure 1, the DLL includes control logic 101, refclk mask generator 102, fbclk mask generator 103,
15 counter 105, delay line 110, and a cell which includes phase detector 181, charge pump 182, Nbias generator 183, and startup circuitry 184. In one embodiment, the DLL receives inputs, such as, for example, ckin 143, weaklocken 142, and programmable_bw 141. Output of the DLL (not shown) includes outputs from delay line 110 including fbclk 162.

 In one embodiment, the aforementioned units are shown as discrete devices. Other
20 embodiments are possible where some or all of these units are integrated within a device or within other devices. In other embodiments, the aforementioned units are distributed throughout a system in hardware, software, or some combination thereof.

 In one embodiment, phase detector 181, charge pump 182, Nbias generator 183, startup circuitry 184, and delay line 110 are components of a conventional DLL. A DLL
25 may also include loop filters.

 In one embodiment, delay line 110 receives an incoming clock (i.e., ckin 143) and delays the signal using cascaded delay stages to generate a set of equally spaced taps. The delay through the individual delay elements is controlled using bias voltages (pbias 187 and nbias 188). In one embodiment, the equal phase distance is possible by controlling
30 each delay stage in delay line 110 using a same control tap. In one embodiment, delay line 110 generates multiphase signals based on an input given (for example, if the clock cycle is 400 ps, delay line 110 generates 8 signals with phases equally spaced about 50 ps per phase).

In one embodiment, two nominally identical taps (taps delayed by a full reference clock cycle, e.g., refclk 161 and fbclk 162) are fed into phase detector 181 (as refclk_pfd 155 and fbclk_pfd 156). Phase error between the nominally identical taps is integrated by charge-pump 182 and a loop filter capacitor to generate bias voltages. A negative feedback
5 loop adjusts the control voltages (and hence the delay of the individual buffers of delay line 110) in a direction that reduces phase error.

In one embodiment, phase detector 180 (or phase and frequency detector) generates Up 185 and Down 186 signals in response to the phase comparison between the reference clock (refclk 161) and feedback clock (fbclk 162). Up 185 and Down 186 drive
10 charge pump 182 which generates a voltage.

In one embodiment, Nbias generator 183 generates nbias voltages based on pbias voltages from charge pump 182. During an initial setup, startup circuitry 184 pulls pbias 187 down to zero. This causes nbias 188 to increase to the highest value. Delay line 110 is set at the fastest setting if the nbias 188 is at its highest value. After a certain length of
15 time, phase detector 181 is enabled and bias voltages are released. The feedback loop tries to lock to the reference frequency.

In one embodiment, a DLL includes at least two bias voltage cells. The values of pbias 187 is in the range of -0.25 to 0.65 V. The value of nbias 188 is in the range of 0.35-0.65 V. In one embodiment, when the circuit is off, pbias 187 is pulled up to VDD and
20 nbias 188 is pulled down to VSS (0). In one embodiment, a higher value of pbias 187 slows down delay line 110, whereas a higher value of nbias 188 speeds up delay line 110.

In one embodiment, a DLL includes control logic 101 to enter and to exit a weak locked state (one of the low power modes). Control logic 101 generates programmable_bw 141 and weaklocken 142. Control logic 101 puts the feedback loop in a weakly locked
25 state (referred herein as a weaklock mode).

Low Power Operation

In one embodiment, control logic 101 lowers the power consumed during periods of inactivity by shutting down delay line 110 while maintaining the bias voltages close
30 enough to the locked values to enable a fast wakeup and relock. During the weaklock mode, delay line 110 is shut down (completely) while identical signals (a clock signal) at a lower clock rate are sent to phase detector 181 to emulate the reference clock input (refclk) and feedback clock input (fbclk) of the DLL under locked conditions. In one embodiment,

both the reference clock input (refclk) and the feedback clock input (fbclk) are disabled during weaklock mode. In one embodiment, depending on circuit architecture, charge pump 182 and phase detector 181 are powered down.

In one embodiment, when the clock to delay line 110 is shut down and the feedback loop is disabled, the bias voltage leaks towards one of the voltage supplies because it is no longer actively driven. In order to support a fast relock, both pbias 187 and nbias 188 are maintained very close to the ideal closed-loop value even when the DLL is partially or fully shutdown. In one embodiment, the bias voltages are periodically refreshed.

In one embodiment, control logic 101 determines whether to provide a clock signal to both the reference input and the feedback input (refclk_pfd 155 and fbclk_pfd 156) of phase detector 181, so that both inputs receive an identical source of input. Control logic 101 shuts down delay line 110 and other unnecessary circuits while leaving the biases on. Control logic 110 feeds an alternate clock to emulate a locked system and the alternate clock is of a lower frequency than ckin 143. In one embodiment, control logic 101 controls the source of refclk_pfd 155 and fbclk_pfd 156 by controlling programmable_bw 141 and weaklocken 142.

In one embodiment, counter 105 generates a clock signal (e.g., ckwklock 160). Control logic 101 sets refclk_pfd 155 and fbclk_pfd 156 (by using programmable_bw 141 and weaklocken 142) to receive the clock signal to emulate a lock condition. For example, if weaklocken 142 is asserted, ckwklock 160 becomes the source to phase detector 181. Phase detector 181 receives identical signals from refclk_pfd 155 and fbclk_pfd 156 with no phase error (emulating a locked condition). Phase detector 180 generates identical Up 185 and Down 186. Charge pump 182 turns on an equal amount of currents on the bias voltages, thus holding the bias voltages steady.

In one embodiment, counter 105 generates ckwklock 160 which is of a lower frequency compared to the clock used during normal operation. In one embodiment, other approaches would be used to disable phase detector 181, charge pump 182, or both.

Refclk mask generator 102 and fbclk mask generator 103 will be described in further detail below with additional references to the remaining figures.

Bias Voltage

In one embodiment, when the DLL are shut down for extended periods of time,

there could be a drift in the bias voltage (and an associated phase error between refclk 161/fbclk 162). In one embodiment, a drift in the bias voltages are corrected (or reduced) by periodically locking the DLL using the original refclk 161 and fbclk 162 from delay line 110. In one embodiment, control logic 101 periodically turns on the DLL to maintain bias voltages (e.g., relock every 500 ns). The operation is also referred to herein as a fast wakeup or a relock. In one embodiment, the DLL performs a fast wakeup for a few clock cycles before going back into the weaklock mode. For example, control logic 101 turns on the DLL (including phase detector 181 and charge pump 182) at a programmable interval and provides refclk 161 and fbclk 162 for one or more clock cycles to prevent a significant drift or to maintain voltage levels held by the two bias voltage cells. In one embodiment, depending on circuit architecture, charge pump 182 and phase detector 181 are powered down.

In one embodiment, control logic 101 provides a clock to delay line 110 and then, in a subsequent clock cycle, disables delay line 110 but enables the inputs to phase detector 181. As such, the relock process is performed for one clock cycle.

In one embodiment, the bias voltage is close to the correct value and therefore the number of clock cycles required to refine the bias voltage is small. In one embodiment, the exit time from the weakly locked state is about 10 clock cycles.

For example, the DLL is locked when pbias 187 is at 0.5 V. In one embodiment, if the total leakage towards Vcc is higher than the total leakage towards Vss, the value of pbias 187 begins to drift upwards. After about a period of time (e.g., 200 ns), depending on the amount of leakage, pbias 187 may be at 0.51 V. Control logic 101 de-asserts weaklocken 142 so that the DLL attempts to relock. In a few clock cycles, the DLL is able to relock and the value of pbias 187 is brought back to 0.5 V.

In one embodiment, the DLL is able to relock (fast recovery) within 1 ns. The power saving is about 60-70% in some embodiments. This enables the DLL to be powered down even though other circuitry is powered down for only a short period, for example, less than 25 ns. Without the weaklock mode, a DLL, in order to relock, needs to charge up the control voltage and to use a comparator to stop the chargeup process.

Figure 2 is a waveform diagram for a weaklock operation in accordance with one embodiment of the invention. Referring to Figure 2, the signals include ckin 201, wklocken 202, refclk 203, refclk_mask 204, fbclk 205, fbclk_mask 206, refclk_pfd 207, and fbclk_pfd 208. In one embodiment, the signals correspond to ckin 143, wklocken 142,

refclk 161, refclk_mask 151, fclk 162, fclk_mask 152, refclk_pfd 155, and fclk_pfd 156, respectively (with respect to Figure 1).

In one embodiment, for example, when exiting the weaklock mode, delay line 110 is enabled. Note that refclk 161 is one cycle ahead of fclk 162. If the inputs to phase
5 detector 181 are switched from ckwklock 160 to refclk 161 and fclk 162 immediately upon exit from the weaklock state, phase detector 180 may detect a phase error (e.g., false phase error 220) of almost a clock cycle from the first refclk 203 edge to the corresponding edge of fclk 205. Phase detector 181 may mistakenly attempt to reduce the delay because of the false phase error 220.

10 In one embodiment, refclk mask generator 102 generates a signal (e.g., refclk_mask 151) to inhibit (mask) refclk pulses (e.g., 211) when the DLL exits from a weaklock mode. In one embodiment, refclk_mask 204 and fclk_mask 206 ensure that phase error information is retained during the wakeup to support fast relock.

In one embodiment, refclk_mask generator 102 generates a mask signal to inhibit
15 signal 163 for at least one clock cycle. Signal 163 (with respect to Figure 1) is the source of refclk 161 if weakloaden 142 is not asserted. In one embodiment, refclk_mask 204 at least masks the first clock cycle of the refclk 203 after exiting from the weaklock mode. In other embodiment, two or more clock cycle of refclk 203 are inhibited.

In one embodiment, depending on the amount of drift in the bias voltages, there
20 exists a risk of potential glitches when switching to different clock signals. This risk of glitches is avoided by generating refclk_mask 204 and fclk_mask 206 based on the negative edges of refclk 203 and fclk 205 respectively (rather than based on ckin 201).

Figure 3 is a waveform diagram for a programmable frequency operation in accordance with one embodiment of the invention. Referring to Figure 3, the signals
25 include ckin 301, programmable_bw 302, refclk 303, refclk_mask 304, fclk 305, fclk_mask 306, refclk_pfd 307, and fclk_pfd 308. In one embodiment, the signals correspond to ckin 143, programmable_bw 141, refclk 161, refclk_mask 151, fclk 162, fclk_mask 152, refclk_pfd 155, and fclk_pfd 156 respectively (with respect to Figure 1).

In one embodiment, the DLL provides programmability over loop bandwidth by
30 using digital control of the update rate of phase detector 181 during normal operation. A same charge pump similar to charge pump 182 is used (no new design is required) by using the programmable bandwidth mechanism.

For example, to reduce the frequency into half, refclk_mask 304 and fclk_mask

306 are generated such that every alternate refclk 303 and fbclk 305 are inhibited (e.g., pulse 310 and pulse 311). The result is shown as refclk_pfd 307 and fbclk_pfd 308 in Figure 4. In one embodiment, the DLL is reusable for generating different bandwidths as required.

5 In one embodiment, refclk_mask generator 102 and fbclk_mask generator 103 generate mask signals to inhibit refclk 303 and fbclk 305 (respectively) for a plurality of clock cycles in order to control the frequencies of the signals.

Figure 4 is a flow diagram of one embodiment of a process for a low power delay-locked loop. The process is performed by processing logic that may comprise hardware
10 (circuitry, dedicated logic, etc.), software (such as one that is run on a general purpose computer system or a dedicated machine), or a combination of both. In one embodiment, the process is performed in conjunction with a DLL with respect to Figure 1. In one embodiment, the process is performed in conjunction with a computer system with respect to Figure 5. In one embodiment, the process is performed in conjunction with a low power
15 PLL.

Referring to Figure 4, in one embodiment, processing logic begin by generating a periodical signal (process block 401). Processing logic determines whether to set both a reference input and a feedback input of a phase detector to receive the same periodic signal (process block 402). In one embodiment, in response to a power-saving mode,
20 processing logic determines whether to use the periodic signal as inputs to a phase detector when delay cells are powered down to save power.

In one embodiment, processing logic regulates the delay period of multiple delay stages in a delay line (process block 403). In one embodiment, the delay period is based at least on a phase difference generated by the phase detector.

25 In one embodiment, processing logic determines whether to perform a fast relock periodically (process block 404). For example, the fast relock is performed for two or more clock cycles to maintain voltage levels held by bias voltage cells. Processing logic sets the reference input and the feedback input to receive a signal from the first delay cell and a signal from a nominal delay cell respectively during the fast relock.

30 In one embodiment, processing logic generates a mask signal to inhibit a periodic signal received at the reference input for at least one clock cycle (process block 405). In one embodiment, the periodic signal received at the reference input is approximately one clock cycle ahead of the periodic signal received at the feedback input.

In one embodiment, processing logic determines whether to generate mask signals to inhibit periodic clock signals for a plurality of clock cycles, to control the frequencies of output signals (process block 406).

Figure 5 illustrates an example of a computer system in conjunction with one
5 embodiment of the invention. Processor 705 accesses data from level 1 (L1) cache memory 706, level 2 (L2) cache memory 710, and main memory 715. In one embodiment, cache memory 710 is a shared cache for more than one processor core.

In one embodiment, memory/graphic controller 716, IO controller 717, or combinations thereof is integrated in processor 705. In one embodiment, parts of
10 memory/graphic controller 716, parts of IO controller 717, or combinations thereof is integrated in processor 705.

Processor 705 may have any number of processing cores. Other embodiments of the invention, however, may be implemented within other devices within the system or distributed throughout the system in hardware, software, or some combination thereof.

15 Main memory 715 may be implemented in various memory sources, such as dynamic random-access memory (DRAM), hard disk drive (HDD) 720, solid state disk 725 based on NVRAM technology, or a memory source located remotely from the computer system via network interface 730 or via wireless interface 740 containing various storage devices and technologies. The cache memory may be located either within
20 the processor or in close proximity to the processor, such as on the processor's local bus 707. Furthermore, the cache memory may contain relatively fast memory cells, such as a six-transistor (6T) cell, or other memory cell of approximately equal or faster access speed.

Other embodiments of the invention, however, may exist in other circuits, logic
25 units, or devices within the system of Figure 5. Furthermore, in other embodiments of the invention may be distributed throughout several circuits, logic units, or devices illustrated in Figure 5.

The invention is not limited to the embodiments described, but can be practiced with modification and alteration within the spirit and scope of the appended claims. For
30 example, it should be appreciated that the present invention is applicable for use with all types of semiconductor integrated circuit ("IC") chips. Examples of these IC chips include but are not limited to processors, controllers, chipset components, programmable logic arrays (PLA), memory chips, network chips, or the like. Moreover, it should be

appreciated that exemplary sizes/models/values/ranges may have been given, although embodiments of the present invention are not limited to the same. As manufacturing techniques (e.g., photolithography) mature over time, it is expected that devices of smaller size could be manufactured.

5 Whereas many alterations and modifications of the embodiment of the present invention will no doubt become apparent to a person of ordinary skill in the art after having read the foregoing description, it is to be understood that any particular embodiment shown and described by way of illustration is in no way intended to be considered limiting. Therefore, references to details of various embodiments are not
10 intended to limit the scope of the claims which in themselves recite only those features regarded as essential to the invention.

CLAIMS

1. An apparatus for generating signal comprising:
 - a phase detector comprising a reference input and a feedback input to determine a phase difference; and
 - 5 a controller to determine whether to perform a relock for duration of one or more clock cycles to maintain voltage levels held by one or more bias voltage cells.
2. The apparatus of claim 1, further comprising a counter to generate a first signal,
10 wherein the controller is operable to set both the reference input and the feedback input to receive the first signal to maintain the phase difference.
3. The apparatus of claim 1, further comprising a plurality of delay cells comprising a first delay cell and a second delay cell, wherein the controller is operable to cause the reference
15 input and the feedback input to receive an input of the first delay cell and an output of the second delay cell respectively.
4. The apparatus of claim 1, further comprising a plurality of delay cells, wherein the controller determines to perform the relock in response to a power saving mode during
20 which the plurality of delay cells are powered down.
5. The apparatus of claim 1, further comprising a charge pump coupled to control a delay period associated with a plurality of delay cells based on the phase difference information from the phase detector, wherein the controller is operable to determine whether or not to
25 power down the phase detector, the charge pump, or both.
6. The apparatus of claim 1, wherein the controller is operable to cause to perform, periodically, the relock for duration of one or more clock cycles.
- 30 7. The apparatus of claim 1, wherein the controller is operable to generate a first mask signal to inhibit, for at least one clock cycle, a first periodic signal received at the reference input if the first periodic signal is approximately one clock cycle ahead of a second periodic signal received at the feedback input.

8. The apparatus of claim 1, wherein the controller is operable to generate a first mask signal and a second mask signal to inhibit, for a plurality of clock cycles, to control the frequencies of signals which the reference input and the feedback input receive.
- 5
9. The apparatus of claim 2, wherein a frequency of the first signal is lower than an operating frequency of a clock signal which the reference clock input receives under a normal power mode.
- 10
10. A method for generating signal comprising:
- determining, in response to a power-saving mode, whether to perform a relock for a duration of one or more clock cycles to maintain voltage levels held by one or more bias voltage cells; and
- setting a reference input and a feedback input of a phase detector to receive a first
- 15
- periodic signal from a first delay cell and a second periodic signal from a second delay cell respectively, wherein the first and the second delay cells are a part of a plurality of delay cells.
11. The method of claim 10, further comprising determining whether to provide a first
- 20
- signal to both the reference input and the feedback input in response to the power saving mode during which a plurality of delay cells are powered down.
12. The method of claim 10, further comprising controlling a delay period associated with a plurality of delay cells based at least on a phase difference generated by the phase
- 25
- detector.
13. The method of claim 10, further comprising determining whether to perform, periodically, a relock for duration of one or more clock cycles
- 30
14. The method of claim 10, further comprising generating a first mask signal to inhibit, for at least one clock cycle, a first periodic signal received at the reference input if the first periodic signal is approximately one clock cycle ahead of a second periodic signal received at the feedback input.

15. The method of claim 10, further comprising generating a first mask signal and a second mask signal to inhibit, for a plurality of clock cycles, to control frequencies of signals which the reference input and the feedback input receive.
- 5
16. A computing system comprising:
- a processor; and
 - a memory interface comprising a delay-locked loop comprising
 - a phase detector comprising a reference input and a feedback input to
 - 10 determine a phase difference based on the two inputs;
 - a controller to determine whether to perform a relock for duration of one or more clock cycles to maintain voltage levels held by one or more bias voltage cells; and
 - a charge pump coupled to control a delay period associated with a plurality
 - 15 of delay cells, based on the phase difference from the phase detector.
17. The system of claim 16, further comprising a counter to generate a first signal, wherein the controller is operable to set the reference input and the feedback input to receive an
- 20 identical signal which is the first signal in response to a power-saving mode.
18. The system of claim 16, wherein the controller is operable to cause to perform, periodically, a relock for duration of one or more clock cycles.
- 25 19. The system of claim 16, wherein the controller is operable to generate a first mask signal to inhibit, for at least one clock cycle, a first periodic signal received at the reference input if the first periodic signal is approximately one clock cycle ahead of a second periodic signal received at the feedback input.
- 30 20. The system of claim 16, wherein the controller is operable to generate a first mask signal and a second mask signal to inhibit, for a plurality of clock cycles, to control the frequencies of signals which the reference input and the feedback input receive.

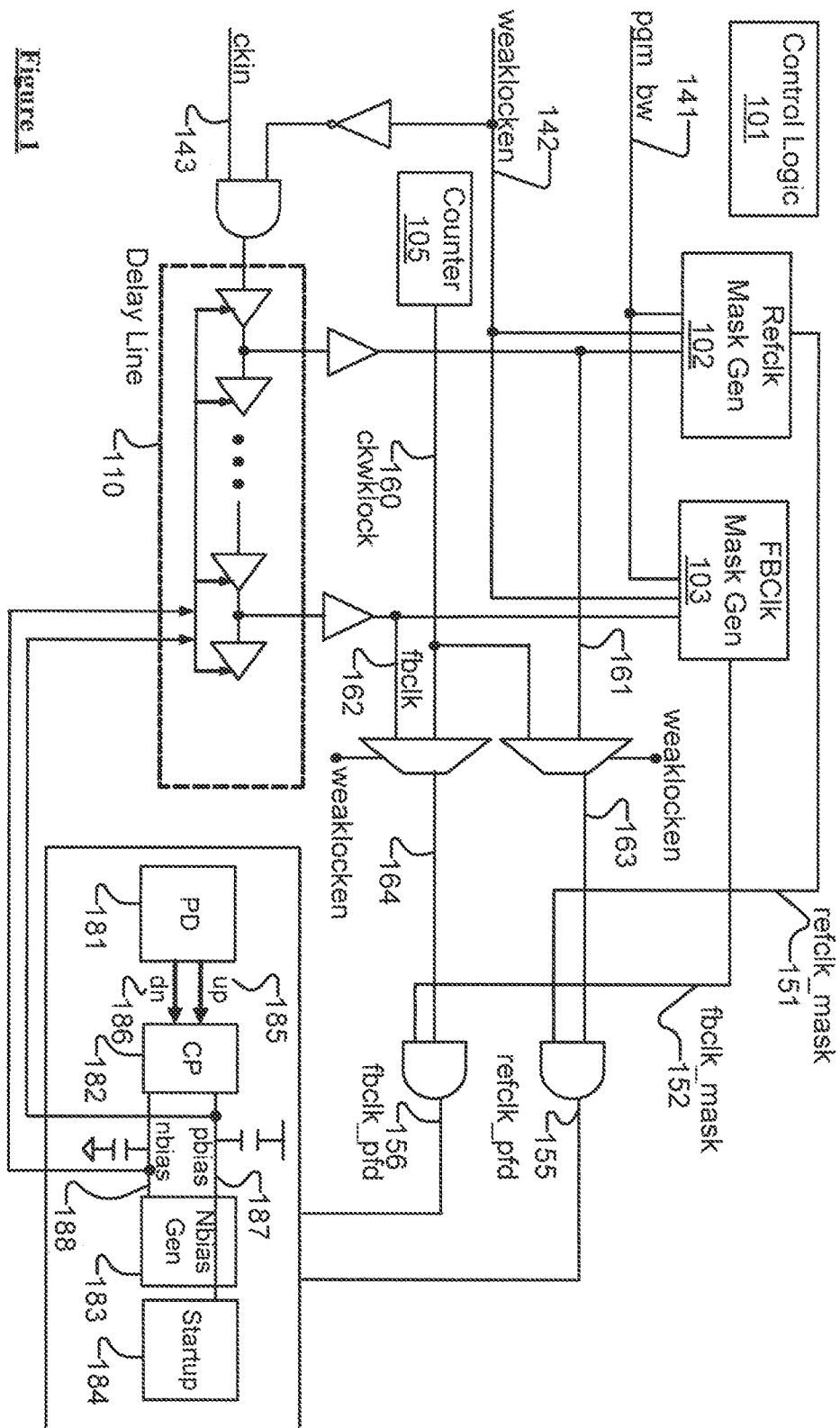


Figure 1

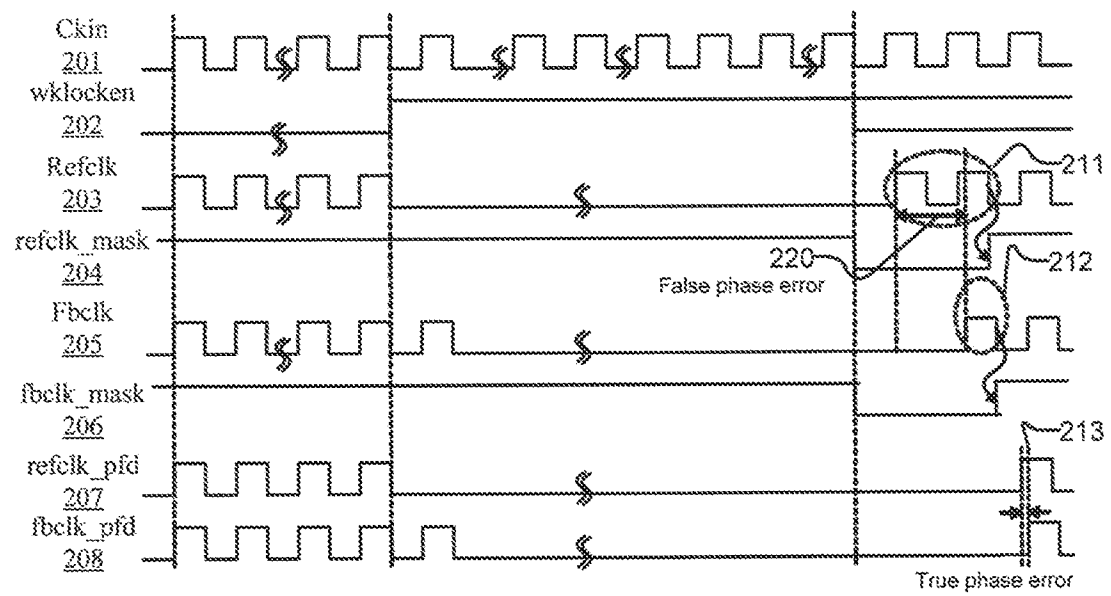
Figure 2

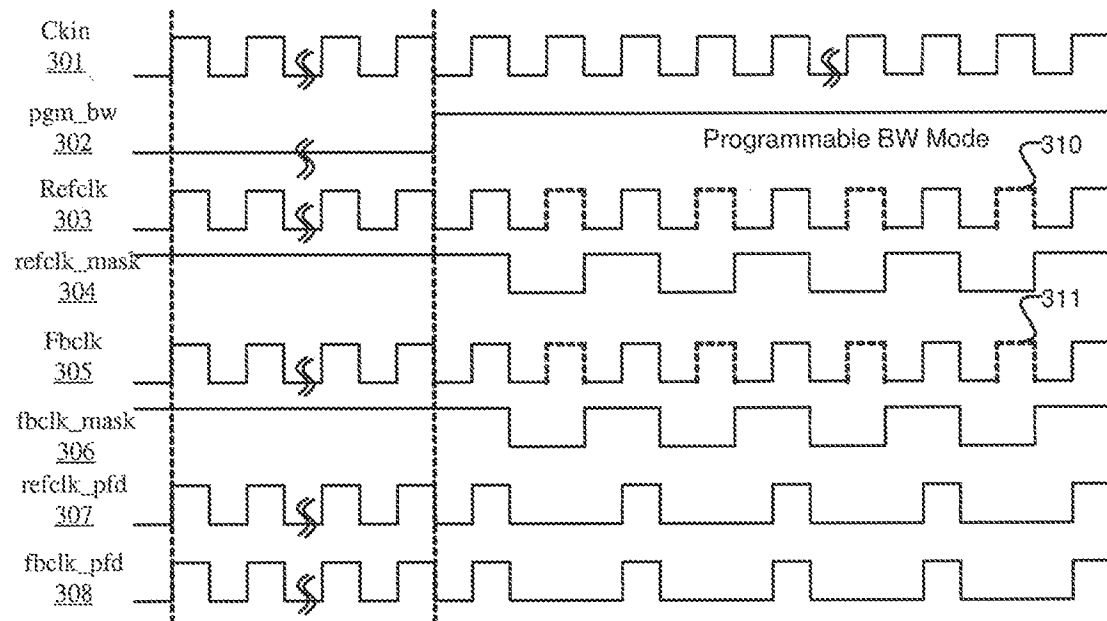
Figure 3

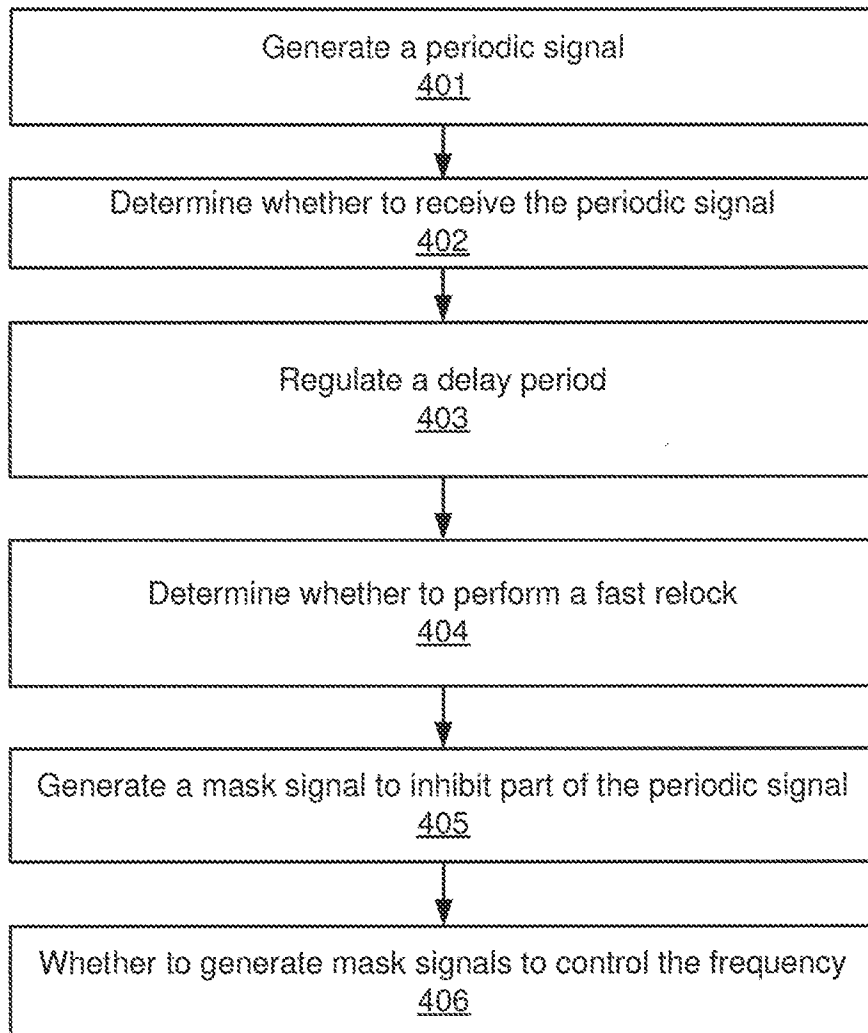
Figure 4

Figure 5