(19) österreichisches patentamt

(10) **AT 522431 A4 2020-11-15**

(12) Österreichische Patentanmeldung

(21) Anmeldenummer: A 50847/2019
(22) Anmeldetag: 07.10.2019
(43) Veröffentlicht am: 15.11.2020

(51) Int. Cl.: ***G11C 7/10*** (2006.01)
H03K 3/037 (2006.01)
H03K 5/135 (2006.01)
H03L 7/00 (2006.01)

(56) Entgegenhaltungen:
US 2009153202 A1
US 5250858 A
WO 0195551 A2
JP 2009033553 A
WO 2011153177 A2
ALIOTO Massimo, CONSOLI Elio, PALUMBO Gaetano: "Chapter 3: Clocked Storage Elements." In: "Flip-Flop Design in Nanometer CMOS", Cham: Springer International Publishing, 2015, S. 57–79, ISBN: 978-3-319-01996-3, DOI: 10.1007/978-3-319-01997-0_3
US 9697309 B1
CN 108199698 A
US 2018241539 A1

(71) Patentanmelder:
Pfaff Markus Dipl.Ing. Dr.
4212 Neumarkt im Mühlkreis (AT)

(72) Erfinder:
Pfaff Markus Dipl.Ing. Dr.
4212 Neumarkt im Mühlkreis (AT)

AT 522431 A4 2020-11-15

(54) SYNCHRONISATIONSSCHALTUNG MIT VERRINGERTER LATENZ UND GESTEIGERTEM DURCHSATZ

(57) Im Folgenden wird ein Verfahren sowie eine Schaltung zur Synchronisation asynchroner digitaler Signale beschrieben. Gemäß einem Ausführungsbeispiel weist die Schaltung eine erste Serienschaltung umfassend ein erstes Eingangs-Flipflop und ein erstes Ausgangs-Flipflop und eine zweite Serienschaltung umfassend ein zweites Eingangs-Flipflop sowie ein zweites Ausgangs-Flipflop auf, wobei der ersten Serienschaltung und der zweiten Serienschaltung ein Eingangssignal zugeführt ist, dessen Pegel einen binären Wert repräsentiert. Die Schaltung weist weiter eine kombinatorische Logikschaltung auf, welche dazu ausgebildet ist, als Eingangswerte einen Ausgangswert der ersten Serienschaltung, einen Ausgangswert der zweiten Serienschaltung und einen gespeicherten alten Wert eines synchronisierten Ausgangssignals zu empfangen und basierend darauf einen aktualisierten Wert des synchronisierten Ausgangssignals zu erzeugen. Das erste Eingangs-Flipflop ist mit einem ersten Taktsignal getaktet, und das zweite Eingangs-Flipflop ist mit einem zweiten Taktsignal getaktet ist, welches in Bezug auf das erste Taktsignal um eine Verzögerungszeit verzögert ist.

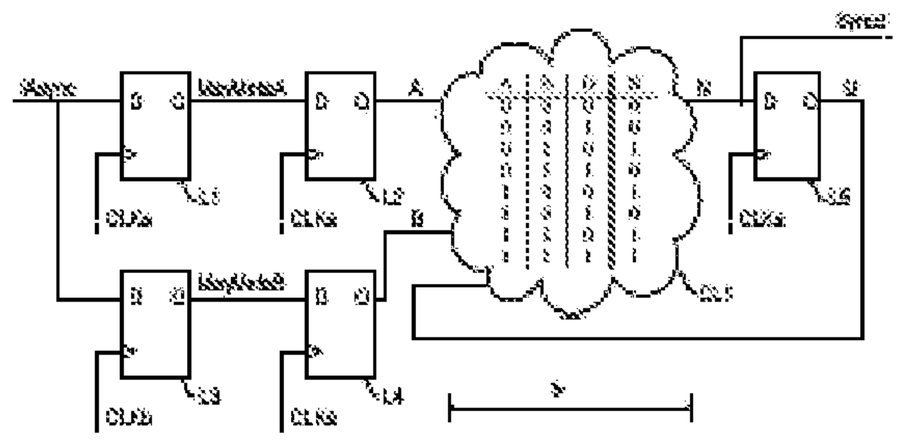

Fig. 5

ZUSAMMENFASSUNG / ABSTRACT

Im Folgenden wird ein Verfahren sowie eine Schaltung zur Synchronisation asynchroner digitaler Signale beschrieben. Gemäß einem Ausführungsbeispiel weist die Schaltung eine erste Serienschaltung umfassend ein erstes Eingangs-Flipflop und ein erstes Ausgangs-Flipflop und eine zweite Serienschaltung umfassend ein zweites Eingangs-Flipflop sowie ein zweites Ausgangs-Flipflop auf, wobei der ersten Serienschaltung und der zweiten Serienschaltung ein Eingangssignal zugeführt ist, dessen Pegel einen binären Wert repräsentiert. Die Schaltung weist weiter eine kombinatorische Logikschaltung auf, welche dazu ausgebildet ist, als Eingangswerte einen Ausgangswert der ersten Serienschaltung, einen Ausgangswert der zweiten Serienschaltung und einen gespeicherten alten Wert eines synchronisierten Ausgangssignals zu empfangen und basierend darauf einen aktualisierten Wert des synchronisierten Ausgangssignals zu erzeugen. Das erste Eingangs-Flipflop ist mit einem ersten Taktsignal getaktet, und das zweite Eingangs-Flipflop ist mit einem zweiten Taktsignal getaktet ist, welches in Bezug auf das erste Taktsignal um eine Verzögerungszeit verzögert ist.

Fig. 5

SYNCHRONISATIONSSCHALTUNG MIT VERRINGERTER LATENZ UND GESTEIGERTEM DURCHSATZ

TECHNISCHES GEBIET

[0001] Die vorliegende Beschreibung betrifft das Gebiet der digitalen Schaltungstechnik, insbesondere eine Schaltung zur Synchronisation asynchroner digitaler Signale.

HINTERGRUND

[0002] Digitale Schaltungen werden nach dem aktuellen Stand der Technik auf Basis des sogenannten synchronen Entwurfsparadigmas konstruiert. Hierbei steuert ein Taktsignal das Laden neuer Inhalte in alle Flipflops bzw. Register des Systems oder Subsystems. Kurze Zeit vor sowie nach dem eigentlichen Einlesezeitpunkt neuer Speicherwerte müssen die Signale an den Eingängen der Speicherelemente stabil bleiben, dürfen sich also nicht ändern. Diese Zeitspannen werden Setup- und Hold-Zeit genannt. Um diese Bedingung stets zu erfüllen, kann beispielsweise mit einem einzigen Taktsignal innerhalb des ganzen Systems gearbeitet werden. Allerdings ist diese Vorgehensweise auf Grund von Nebenbedingungen nicht immer möglich. So erfordern unterschiedliche Aufgaben im Gesamtsystem beispielsweise häufig mehrere Taktsignale mit unterschiedlichen Taktfrequenzen.

[0003] Die auf diese Art entstehenden Bereiche, welche mit unterschiedlichen Taktfrequenzen arbeiten, werden als *Clock-Domains* bezeichnet. Diese Clock-Domains sind typischerweise asynchron zueinander, d.h. die Taktsignale stehen nicht in einem festen Phasenbezug zueinander. Daher werden die oben angesprochenen Setup- und Hold-Zeiten von Signalen, die von einer in eine andere Clock-Domain übergehen, unter Umständen verletzt. Diese Situation kann nur unter bestimmten, häufig jedoch nicht praktikablen Bedingungen umgangen werden.

[0004] Die gleiche Situation liegt vor, wenn asynchrone Eingangssignale an eine synchron getaktete Schaltung herangeführt werden, was in der Praxis ein ebenfalls häufiger Fall ist. Die Lösung dieses prinzipiellen Problems liegt im Einsatz so genannter Synchronisationsschaltungen. Die am meisten verbreiteten Schaltungen dieser Art sind seit Jahrzehnten bekannt, aber neue Ansätze sind auf Grund von deren nicht immer befriedigenden Eigenschaften bereits seit langem Gegenstand der Forschung. Vorliegend wird ein Vorschlag

für eine solche Schaltung gemacht, welche gegenüber den in der Literatur und Praxis angetroffenen Lösungsansätzen Vorteile bietet.

[0005] Als charakteristische Parameter zur Bewertung von Synchronisationsschaltungen können die Zuverlässigkeit der Schaltung, die Synchronisationslatenz und der Durchsatz verwendet werden. Die Zuverlässigkeit der Schaltung wird zumeist durch die MTBF (*Mean Time Between Failures*) repräsentiert. Die Synchronisationslatenz (im weiteren Verlauf auch als Latenz bezeichnet) gibt die Verzögerung gemessen vom Auftreten einer Signaländerung am Eingang der Schaltung bis zu deren Ankunft innerhalb der Clock-Domain an, zu welcher sie synchronisiert wird. Der Durchsatz gibt die Rate an, mit der Signaländerungen an den Eingang der Synchronisationsschaltung angelegt können, ohne dass eine dieser Änderungen bei der Synchronisation verloren geht.

[0006] Der Erfinder hat es sich zur Aufgabe gemacht, bestehende Schaltungen zur Synchronisation asynchroner Signale im Hinblick auf Latenz und/oder Durchsatz bei akzeptabler MTBF zu verbessern.

ZUSAMMENFASSUNG

[0007] Die oben erwähnte Aufgabe wird durch die Schaltung gemäß Anspruch 1 sowie das Verfahren durch Anspruch 8 gelöst. Verschiedene Ausführungsbeispiele und Weiterentwicklungen sind Gegenstand der abhängigen Ansprüche.

[0008] Im Folgenden wird ein Verfahren sowie eine Schaltung zur Synchronisation asynchroner digitaler Signale beschrieben. Gemäß einem Ausführungsbeispiel weist die Schaltung eine erste Serienschaltung umfassend ein erstes Eingangs-Flipflop und ein erstes Ausgangs-Flipflop und eine zweite Serienschaltung umfassend ein zweites Eingangs-Flipflop sowie ein zweites Ausgangs-Flipflop auf, wobei der ersten Serienschaltung und der zweiten Serienschaltung ein Eingangssignal zugeführt ist, dessen Pegel einen binären Wert repräsentiert. Die Schaltung weist weiter eine kombinatorische Logikschaltung auf, welche dazu ausgebildet ist, als Eingangswerte einen Ausgangswert der ersten Serienschaltung, einen Ausgangswert der zweiten Serienschaltung und einen gespeicherten alten Wert eines synchronisierten Ausgangssignals zu empfangen und basierend darauf einen aktualisierten Wert des synchronisierten Ausgangssignals zu erzeugen. Das erste Eingangs-Flipflop ist mit einem ersten Taktsignal getaktet, und das zweite Eingangs-Flipflop ist mit

einem zweiten Taktsignal getaktet ist, welches in Bezug auf das erste Taktsignal um eine Verzögerungszeit verzögert ist.

[0009] Das Verfahren umfasst gemäß einem Ausführungsbeispiel das Abtasten eines Eingangssignals synchron zu einem ersten Taktsignal und Speichern des abgetasteten Wertes in einem ersten Eingangs-Flipflop sowie das Abtasten des Eingangssignals synchron zu einem zweiten Taktsignal und Speichern des abgetasteten Wertes in einem zweiten Eingangs-Flipflop, wobei das zweite Taktsignal in Bezug auf das erste Taktsignal um eine Verzögerungszeit verzögert ist. Das Verfahren umfasst weiter das Weiterreichen des in dem ersten Eingangs-Flipflop gespeicherten Wertes an ein erstes Ausgangs-Flipflop und Weiterreichen des in dem zweiten Eingangs-Flipflop gespeicherten Wertes an ein zweites Ausgangs-Flipflop und das Verknüpfen der in dem ersten Ausgangs-Flipflop und dem zweiten Ausgangs-Flipflop gespeicherten Werte mit einem gespeicherten alten Wert eines synchronisierten Ausgangssignals mittels einer kombinatorischen Logikschaltung, wobei das Ergebnis der Verknüpfung den neuen Wert des synchronisierten Ausgangssignals repräsentiert.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0010] Nachfolgend werden Ausführungsbeispiele anhand von Abbildungen näher erläutert. Die Darstellungen sind nicht zwangsläufig maßstabsgetreu und die Ausführungsbeispiele sind nicht nur auf die dargestellten Aspekte beschränkt. Vielmehr wird Wert darauf gelegt, die den Ausführungsbeispielen zugrunde liegenden Prinzipien darzustellen.

[0011] Figur 1 illustriert eine Standard-Synchronisationsschaltung aus zwei direkt hintereinander geschalteten Flipflops.

[0012] Figuren 2-4 zeigen anhand von Timing-Diagrammen Beispiele für mögliche Signalverläufe (*signal waveforms*) für die Schaltung aus Fig. 1.

[0013] Figur 5 illustriert eine Synchronisationsschaltung gemäß einem Ausführungsbeispiel.

[0014] Figuren 6 und 7 zeigen anhand von Timing-Diagrammen Beispiele für mögliche Signalverläufe für die Schaltung aus Fig. 5.

[0015] Figur 8 zeigt anhand von Timing-Diagrammen die Signalverläufe in der Schaltung aus Fig. 5 für ein Eingangssignal, wie es bei der Schaltung aus Fig. 1 zur maximalen Latenz führt.

[0016] Figur 9 illustriert die Zeitpunkte einer sicher korrekten Abtastung des Eingangssignals in der Schaltung aus Fig. 5 bei verschiedenen Phasenlagen des Eingangssignals.

DETAILLIERTE BESCHREIBUNG

[0017] Fig. 1 illustriert eine sehr gängige Synchronisationsschaltung zur Synchronisation (in Bezug auf ein Taktsignal) asynchroner Signale. Eine derartige Schaltung ist z.B. in *U. Tietze, C. Schenk, Electronic Circuits, 2nd ed., Springer-Verlag, 2008, ISBN 978-3-540-00429-5, S. 679-680*, beschrieben. Demnach wird das asynchrone Eingangssignal iAsync dem Eingang D eines taktflankengesteuerten D-Flipflops L1 zugeführt. Der Pegel des Eingangssignals iAsync repräsentiert einen binären Wert (d.h. 0 oder 1), der auch als boolscher Zustand („wahr“ oder „falsch“) interpretiert werden kann. Der Wert/Zustand des Eingangssignals iAsync wird bei jeder positiven Flanke des Taktsignals CLKa an den Ausgang Q des D-Flipflops L1 übertragen. Da sich das Eingangssignal auch annähernd gleichzeitig zur positiven Taktflanke ändern kann, können metastabile Zustände im D-Flipflop L1 auftreten. Um fehlerhafte Zustände zu vermeiden wird nicht das am Ausgang Q des ersten D-Flipflop L1 zur Verfügung stehende Signal MayMeta als Ausgangssignal verwendet, sondern dem Ausgang Q des D-Flipflops L1 wird ein weiteres D-Flipflop L2 nachgeschaltet, welches mit demselben Taktsignal CLK getaktet ist. Das heißt, der Ausgang Q des ersten D-Flipflops L1 ist mit dem Eingang D des zweiten D-Flipflops L2 verbunden. Am Ausgang Q des zweiten D-Flipflops L2 steht als Ausgangssignal das synchronisierte Signal Syncd zur Verfügung.

[0018] Die getaktete Schaltung, zu deren Takt das asynchrone Eingangssignal iAsync synchronisiert werden soll, wird als „Taktdomäne“ (*Clock Domain*) bezeichnet. Innerhalb einer Taktdomäne ändern sich Zustände von Signalen synchron zu dem Taktsignal, das zu der jeweiligen Taktdomäne gehört.

[0019] In den Fig. 2 bis 4 sind exemplarisch verschiedene Beispiele von Signalverläufen (*signal waveforms*) dargestellt. Generell repräsentiert das grau schattierte Intervall um die positiven Taktflanken (Zeitpunkte t_0, t_1, etc.) des Taktsignals CLK die Summe aus Setup-

Zeit t_{SETUP} und Haltezeit t_{HOLD}. Das heißt, in Bezug auf die zum Zeitpunkt t_0 auftretende Taktflanke repräsentiert das grau schattierte Intervall das Intervall von $t_0 - t_{SETUP}$ bis $t_0 + t_{HOLD}$. Das gleiche gilt für Fig. 6 und 7.

[0020] Fig. 2 zeigt den Fall minimaler Latenz, wobei ein Zustandswechsel (von Low auf High im vorliegenden Beispiel) des Eingangssignals iAsync am Ende der Hold-Zeit auftritt, d.h. zum Zeitpunkt $t_0 + t_{HOLD}$. In diesem Fall kann das D-Flipflop L1 einen metastabilen Zustand einnehmen und, sofern das D-Flipflop L1 aus der Metastabilität in den neuen Zustand fällt (was nicht notwendigerweise der Fall sein muss), steht der Wert des Eingangssignals zum Zeitpunkt $t_1 = t_0 + T_{CLK}$ in der Taktdomäne des D-Flipflop L2 zur Verfügung. Die Synchronisationslatenz des Ausgangssignals Syncd in Bezug auf das Eingangssginal iAsync ist in diesem Fall etwas kleiner als die Periodendauer T_{CLK} des Taktsignals CLK und beträgt $T_{CLK} - t_{HOLD}$.

[0021] Fig. 3 zeigt den Fall maximaler Latenz, wobei ein Zustandswechsel des Eingangssignals iAsync (im vorliegenden Beispiel von Low auf High) am Anfang der Setup-Zeit auftritt, d.h. zum Zeitpunkt $t_0 - t_{SETUP}$. In diesem Fall kann das D-Flipflop L1 einen metastabilen Zustand einnehmen und, sofern das D-Flipflop L1 aus der Metastabilität in den alten Zustand fällt, steht der aktuelle Zustand erst zwei Perioden später, d.h. zum Zeitpunkt $t_2 = t_0 + 2T_{CLK}$, in der Takt-Domäne des D-Flipflop L2 zur Verfügung. Die Synchronisationslatenz des Signalwechsels des Eingangssignals iAsync bis zur Ankunft des zum Takt synchronisierten Signalwechsels in der Taktdomäne ist in diesem Fall etwas größer als die doppelte Periodendauer $2T_{CLK}$ des Taktsignals CLK, nämlich $2T_{CLK} + t_{SETUP}$. Diese Zeitdauer bezieht sich nur auf die Ankunft des Eingangssignalwechsels in der Takt-Domäne. Dort vergeht nach der steigenden Taktflanke noch eine Zeitdauer t_{DEL}, die Signallaufzeit (*propagation delay*) zwischen Taktflanke und Auftreten des Signalwechsels am Ausgang des D-Flipflops L2. Diese Zeitdauer vergeht jedoch nach der Ankunft des Eingangssignals in der empfangenden Takt-Domäne. Sie wird daher nicht der Synchronisationslatenz zugeschlagen. Aus diesem Grunde wird diese Zeitdauer im Beispiel aus Fig. 2 nicht dargestellt.

[0022] Fig. 4 illustriert mehrere asynchrone Zustandswechsel des Eingangssignals iAsync zur Darstellung des maximal erzielbaren Durchsatzes bei der Synchronisation nach Fig. 1. Der erste Zustandswechsel tritt am Anfang der Setup-Zeit auf, d.h. zum Zeitpunkt $t_0 - t_{SETUP}$. In diesem Beispiel nimmt das D-Flipflop L1 temporär einen metastabilen Zustand an und fällt dann in den alten Zustand zurück, d.h. das Signal MayMeta bleibt in dem

Low-Zustand bis zum Zeitpunkt $t_1 + t_{DEL}$ und der Zustandswechsel wird erst zum Zeitpunkt $t_1 + t_{DEL}$ im Ausgangssignal Syncd „sichtbar“. Kurz nach dem Zeitpunkt t_1 wechselt das Eingangssignal iAsync wieder auf einen Low-Zustand. Dieser kann vom D-Flipflop L1 nicht mehr in derselben Taktperiode erfasst werden und folglich wird dieser Zustandswechsel in dem Signal MayMeta erst zum Zeitpunkt $t_2 + t_{DEL}$ und im Ausgangssignal Syncd erst zum Zeitpunkt $t_3 + t_{DEL}$ sichtbar, etc.

[0023] Von den Eigenschaften der Schaltung aus Fig. 1 werden im Folgenden die Werte für die Latenz und den Durchsatz näher beleuchtet. Die Latenz liegt in einem Bereich von knapp unterhalb einer Taktperiode, d.h. ungefähr $T_{CLK} - t_{HOLD}$, bis zu etwas mehr als zwei Taktperioden, d.h. $2T_{CLK} + t_{SETUP}$. Hierbei wird die steigende Taktflanke des Taktsignals CLK der empfangenden Clock-Domain als Ankunftszeitpunkt betrachtet. Zu diesem Zeitpunkt (vgl. Fig. 4, Zeitpunkte t_1, t_2, etc.) kann das Eingangssignal als synchronisiert betrachtet werden, wenn auch noch eine geringe zusätzliche Verzögerungszeit (Signallaufzeit t_{DEL}) benötigt wird, bis sich dieses synchronisierte Ausgangssignal Syncd auf die nachfolgende kombinatorische Schaltung auswirken kann.

[0024] Der Durchsatz wird durch die Zeitspanne zwischen zwei Signaländerungen bestimmt, die zumindest $T_{CLK} + t_{SETUP} + t_{HOLD}$ (vgl. Fig. 4) betragen muss. Das Taktsignal CLK der empfangenden Clock-Domain muss also gegenüber der (asynchron) sendenden Clock-Domain stets eine höhere Taktfrequenz aufweisen. Die gleiche Taktfrequenz reicht nicht aus. Die MTBF wird durch die Schaltungseigenschaften des D-Flipflop L1 bestimmt, maßgeblich durch die Zeitspanne $T_{CLK} - (t_{SETUP} + t_{DEL})$, welche diesem zur Auflösung eventuell auftretender Metastabilität zur Verfügung steht. Wird diese Zeitspanne überschritten, liegt der metastabile Zustand als Eingangswert am Flipflop L2 an, was wiederum dessen Metastabilität zur Folge haben kann. In diesem Fall wäre die Synchronisation nicht erfolgreich.

[0025] Die im Folgenden beschriebenen Ausführungsbeispiele betreffen ein Konzept, bestehende Schaltungen zur Synchronisation asynchroner Signale im Hinblick auf Latenz und/oder Durchsatz bei akzeptabler MTBF zu verbessern. Die im Folgenden beschriebene Vorgehensweise kann die erwähnte Verbesserung ermöglichen. Dabei wird zunächst detektiert, ob Metastabilität innerhalb der Synchronisationsschaltung vorliegt. Diesbezüglich werden später zwei Beispiele erläutert.

[0026] Abhängig davon, ob ein metastabiler Zustand detektiert wurde, kann auf den korrekten Zustand rückgeschlossen werden. Binäre Signale, wie sie in digitalen Schaltungen verwendet werden, weisen nur zwei Zustände/Werte auf, nämlich 0 (Low-Zustand) oder 1 (High-Zustand). Liegt Metastabilität vor, so kann diese nur durch eine Änderung des Zustands des Eingangssignals entweder von 0 nach 1 oder aber von 1 nach 0 hervorgerufen worden sein. Wenn also Metastabilität detektiert wurde, dann ist die Negation des zuletzt gültigen Zustands des synchronen Ausgangssignals (in der synchronen Taktdomäne) der korrekte Wert für die weitere Verwendung. Desgleichen gilt dies in trivialer Weise für das Vorliegen eines Signalwechsels, der keine Metastabilität hervorgerufen hat. Die vorgeschlagene Vorgehensweise ermittelt also aus dem alten Zustand (des synchronisierten Ausgangssignals) und der Feststellung, ob Metastabilität bzw. ein Signalwechsel vorliegen oder nicht, einen gültigen neuen taktsynchronen Zustand.

[0027] Das Vorliegen von Metastabilität (bzw. eines Signalwechsels) kann durch Vergleich der Werte aus zwei Synchronisationsketten (je zwei D-Flipflops hintereinander nach Fig. 1) erkannt werden, wenn das erste D-Flipflop einer der Ketten einen um den Wert $t_{SETUP} + t_{HOLD}$verzögerten Takt erhält. Diese Maßnahme garantiert, dass stets eine der beiden Synchronisationsketten das korrekte Signal am Ausgang trägt, wobei aber nicht detektierbar ist, welche der beiden Synchronisationsketten das korrekte Signal liefert. Sind diese beiden Ausgangssignale verschieden, so liegt Metastabilität (bzw. ein Signalwechsel) vor. Diese Vorgehensweise ist mittels an sich bekannter Techniken (Integration mittels *Field Programmable Gate Array*, FPGA, oder als *Application Specific Integated Circuit*, ASIC) leicht umsetzbar und wird im Folgenden anhand des Beispiels aus Fig. 5 noch genauer erläutert.

[0028] Die Schaltungsanordnung aus Fig. 5 umfasst zwei Synchronisationsketten (D-Flipflops L1 und L2 und D-Flipflops L3 und L4), von denen jede aufgebaut ist wie die Schaltung aus Fig. 1. Die D-Flipflops L1 und L2 der ersten Synchronisationskette und das D-Flipflop L4 der zweiten Synchronisationskette werden mit dem Taktsignal CLKa getaktet, zu dem synchronisiert werden soll. Das D-Flipflop L3 der zweiten Synchronisationskette wird mit dem gegenüber CLKa verzögerten Taktsignal CLKb getaktet. Beide Taktsignale CLKa und CLKb haben demnach dieselbe Taktfrequenz $T_{CLK}{}^{-1}$, wobei das Taktsignal CLKb gegenüber dem Taktsignal CLKa um einen kleinen Bruchteil ΔT einer Taktperiode T_{CLK} verzögert ist (wobei $\Delta T \geq t_{SETUP} + t_{HOLD}$). Gemäß einem Beispiel ist die

Verzögerung ΔT zwischen den Taktsignalen CLKa und CLKb nur geringfügig größer als die Summe $t_{SETUP} + t_{HOLD}$. Das Taktsignal CLKb kann mit an sich bekannten Methoden aus dem Taktsignal CLKa abgeleitet werden (z.B. mittels einer *Delay-Locked Loop*, DLL, einer *Phase-Locked Loop*, PLL, oder verschiedensten Verzögerungsschaltungen).

[0029] Gemäß Fig. 5 wird das (asynchrone) Eingangssignal iAsync dem ersten D-Flipflop beider Synchronisationsketten, also den Flipflops L1 und L3 zugeführt. Die Ausgangssignale der Flipflops L1 und L3 werden mit MayMetaA bzw. MayMetaB bezeichnet, und diese sind den Eingängen der nachfolgenden Flipflops L2 bzw. L4 zugeführt. Die Ausgangssignale der Synchronisationsketten (d.h. der Flipflops L2 und L4) sind mit A und B bezeichnet. Aufgrund der Verzögerung des Taktsignals CLKb, das für das erste Flipflop L3 der zweiten Synchronisationskette (Flipflops L3 und L4) verwendet wird, bedeuten ungleiche Zustände der Ausganssignale A und B (d.h. $A \neq B$) notwendigerweise, dass in einer der beiden Flipflops L1 oder L3 entweder Metastabilität oder eine einfache Signaländerung aufgetreten ist, wobei ohne weitere Information nicht bekannt ist, welches der beiden Ausgangssignale, A oder B, das Eingangssignal iAsync korrekt repräsentiert.

[0030] Die „Rekonstruktion“ des korrekten Zustands (Logikpegels) des Eingangssignals iAsync erfolgt mit einer sequenziellen Logik (*sequential Logic*, Schaltnetz), welche eine kombinatorische Logik CL1 (*combinational Logic*, Schaltwerk) und ein weiteres D-Flipflop L5 aufweist, dessen Eingang D das Ausgangssignal N der kombinatorischen Logik CL1 zugeführt ist. Das D-Flipflop L5 ist ebenfalls mit dem Taktsignal CLKa getaktet. Als Eingangssignale empfängt die kombinatorische Logik CL1 die Ausgangssignale A, B der beiden Synchronisationsketten und das rückgekoppelte Ausgangssignal O des weiteren D-Flipflop L5.

[0031] Die Funktionsweise der kombinatorische Logik CL1 ist in Fig. 5 anhand einer Wahrheitstabelle wiedergegeben. In Worte gefasst ist deren Aussage: Wenn die Zustände der Signale A und B gleich sind (was darauf hinweist, dass keine Metastabilität vorliegt), dann gibt die kombinatorische Logik CL1 diesen Zustand an das Flipflop L5 weiter ($N = A = B$). Wenn die Zustände der Signale A und B ungleich sind (was auf Metastabilität oder eine Signaländerung auf dem Eingangssignal hinweist), dann gibt die kombinatorische Logik CL1 das invertierte Ausgangssignal O an das Flipflop L5 weiter ($N = \bar{O}$). Als kombinatorische Gleichung ausgedrückt entspricht der Zustand des Ausgangssignals N Ausgang der kombinatorische Logik CL1 folgender Gleichung:

$$Syncd = N = A \cdot B + A \cdot \neg O + B \cdot \neg O \tag{1}$$

wobei „+“ eine ODER, „·“ eine UND-Verknüpfung und „¬“ eine Negation bezeichnet.

[0032] Eine Vorgehensweise gemäß Fig. 5 zur Detektion von Metastabilität und – falls Metastabilität auftritt – zur Ermittlung des korrekten Zustands des Eingangssignals iAsync erlaubt eine schnelle und schaltungstechnisch effiziente Implementierung (z.B. in einem FPGA). Die Entscheidung, ob Metastabilität vorliegt, kann sehr schnell getroffen werden.

[0033] Wie erwähnt kann aufgrund der Verzögerung ΔT zwischen den Taktsignalen CLKa und CLKb (Taktversatz, *Clock Offset*) bei einer Änderung des Zustands des Eingangssignals iAsync nur eines der beiden Eingangs-Flipflops der beiden Synchronisationsketten (d.h. das Flipflop L1 oder das Flipflop L3) metastabil werden. Diese Zustandsänderung kann entweder ein Übergang von 0 auf 1 oder von 1 auf 0 sein. Im Folgenden wird allgemein ein Zustandswechsel von OldVal auf NewVal betrachtet. Nach einer Taktflanke des Taktsignals CLKb und einer, nur statistisch zu beschreibenden Zeitdauer zum Abklingen von Metastabilität repräsentiert also eines der beiden Ausgangssignale, MayMetaA oder MayMetaB, den korrekten, neuen Zustand NewVal des Eingangssignals iAsync, wohingegen das jeweils andere Ausgangsignal im Falle von Metastabilität einen unzuverlässigen Zustand aufweist (d.h. der Zustand kann korrekt sein, kann aber auch inkorrekt sein). Jenes D-Flipflop (L1, respektive L3), welches an seinem Ausgang Q den korrekten Zustand des Eingangssignals iAsync ausgibt wird im Folgenden als „Sicher-Korrekt-Flipflop“ bezeichnet (*Certainly Correct Flipflop*), wohingegen das D-Flipflop (L3, respektive L1) in der anderen Synchronisationskette als „Zweifelhafter-Zustand-Flipflop“ (*Dubious State Flipflop*) bezeichnet wird.

[0034] Wie erwähnt ist es a priori nicht bekannt, welches der beiden Synchronisationsketten das Sicher-Korrekt-Flipflop enthält. Welches der beiden Flipflops – L1 oder L3 – die Rolle des Sicher-Korrekt-Flipflop übernimmt, hängt von der zeitlichen Beziehung zwischen dem Zeitpunkt der (asynchronen) Zustandsänderung des Eingangssignals iAsync und der steigenden Taktflanke im Taktsignal CLKa ab. Jedenfalls hat das Sicher-Korrekt-Flipflop kurz nach der steigenden Flanke des verzögerten Taktsignals CLKb (und damit auch nach der kurz davor auftretenden Flanke des Taktsignals CLKa) den korrekten Zustand NewVal des Eingangssignals iAsync gespeichert. Für das Zweifelhafter-Zustand-

Flipflop gibt es jedoch zwei Möglichkeiten. Erstens kann es den den richtigen Zustand NewVal annehmen oder metastabil werden und sich bei der Auflösung der Metastabilität für den richtigen Zustand entscheiden. In diesem Fall hat es denselben Zustand wie das Sicher-Korrekt-Flipflop (nämlich NewVal). Zweitens kann es sich (z.B. aufgrund der Verletzung von t_{SETUP} oder t_{HOLD} oder der Phasenlage der Signaländerung des Eingangssignals iAsync relativ zur Taktflanke) für den falschen Wert OldVal entscheiden oder aber metastabil werden und sich bei der Auflösung der Metastabilität für den alten (und falschen) Wert OldVal entscheiden. In diesem Fall hat es einen anderen Wert als das Sicher-Korrekt-Flipflop (nämlich OldVal). Der Zustand der Flipflops L1 und L3 manifestiert sich durch den Zustand/den Pegel der zugehörigen Ausgangssignale MayMetaA bzw. MayMetaB.

[0035] Die von den D-Flipflops L1 und L3 ausgegebenen Zustände (d.h. die Pegel der Signale MayMetaA und MayMetaB) werden an die nachgeschalteten D-Flipflops L2 bzw. L4 weitertransportiert und bei der nächsten Flanke des Taktsignals CLKa von diesen eingelesen und gespeichert. Hierbei besteht nur eine geringe Wahrscheinlichkeit für das Auftreten von Metastabilität, wenn genug Zeit zum Abklingen der möglicherweise in Flipflop L1 oder L3 vorhandenen Metastabilität verfügbar war. Die verfügbare Zeitspanne ist ein wichtiger Faktor bei der Bestimmung der MTBF.

[0036] Die kombinatorische Logikschaltung CL1, welche mit den Ausgängen Q der D-Flipflops L2 und L4 verbunden ist, ist wie oben schon erwähnt in der Lage, unter Berücksichtigung des zuletzt ausgegebenen Zustands/Pegels O (gespeichert in D-Flipflop 5) den neuen synchronisierten Zustand des Ausgangssignals Syncd zu ermitteln. Sind die Pegel der Signale MayMetaA und MayMetaB gleich, so ist ihr Wert der neue Pegel NewVal des Ausgangssignals Syncd, da ja eines der beiden D-Flipflops L1 und L3 sicher den korrekten Wert enthält. In der in Fig. 5 gezeigten Wahrheitstabelle entspricht dieser Fall den beiden oberen und den beiden unteren Zeilen der Tabelle. Sind die Pegel der Signale MayMetaA und MayMetaB verschieden, so enthält eines der D-Flipflops L1 und L3 den korrekten und das andere den inkorrekten Wert. Da dieser Fall nur durch eine Änderung des Pegels des Eingangssignals iAsync eintreten konnte, muss der im D-Flipflop L5 gespeicherte alte Wert OldVal der inkorrekte Wert sein, und folglich kann der korrekte neue Wert NewVal durch Negation des im Flipflop L5 gespeicherten Wertes ermittelt werden. Diese Fälle werden durch die mittleren vier Zeilen der erwähnten Wahrheitstabelle repräsentiert (siehe auch die zur Tabelle äquivalente Gleichung 1).

[0037] Das Ausführungsbeispiel gemäß Fig. 5 hat trotz moderaten Implementationsaufwands im Hinblick auf Latenz und Durchsatz gegenüber der Schaltung aus Fig. 1 signifikant bessere Eigenschaften. Die Latenz bis zur Ankunft in der Empfangstakt-Domäne bewegt sich im Bereich von knapp unter einer Taktperiode (T_{CLK}) bis knapp unter zwei Taktperioden ($t_{SETUP} + (T_{CLK} - \Delta T) + T_{CLK}$). Die maximale Latenz ist damit besser als bei der Schaltung aus Fig. 1. Dies wirkt sich positiv auf die Reaktionszeiten auf asynchron auftretende Ereignisse aus. Unter gewissen Umständen kann die Tatsache, dass die maximale Latenz etwas weniger als zwei Taktzyklen $2T_{CLK}$ beträgt, ein entscheidender Vorteil sein. Auf die Latenzzeiten wird später unter Bezugnahme auf Fig. 6 und 7 noch näher eingegangen.

[0038] Bezüglich der Zeitangaben zur Latenz gilt das bereits bei der Diskussion der Fig. 2-4 Gesagte. Im Anwendungsfall ist im vorliegenden Beispiel auch die Signallaufzeit t_F für die kombinatorische Schaltung CL1 (siehe Fig. 5) innerhalb der empfangenden Takt-Domäne zu berücksichtigen, wobei diese aber nicht der Synchronisationslatenz zuzuschlagen ist. Fig. 6 zeigt den Fall minimaler Latenz. Ein Zustandswechsel (von Low auf High, d.h. von 0 auf 1, im vorliegenden Beispiel) des Eingangssignals iAsync tritt am Ende der Hold-Zeit auf, d.h. zum Zeitpunkt $t_0 + t_{HOLD}$. In diesem Fall kann das D-Flipflop L1 einen metastabilen Zustand einnehmen und, sofern das D-Flipflop L1 in den neuen Zustand wechselt (was nicht notwendigerweise der Fall sein muss), steht der aktuelle Zustand ungefähr eine Periode später, d.h. zum Zeitpunkt $t_1 = t_0 + T_{CLK}$, am Ausgang des D-Flipflop L2 zur Verfügung (Signal A). Die Latenz bis zum Eintreffen des Signalwechsels auf dem Eingangssignal iAsync in der Takt-Domäne von CLKa ist in diesem Fall etwas kleiner als die Periodendauer T_{CLK} des Taktsignals CLK und beträgt $T_{CLK} - t_{HOLD}$. Die Signalände-rung liegt synchron innerhalb der Takt-Domäne von CLKa um die Zeitdauer $t_{DEL} + t_F$ nach der steigenden Flanke von CLKa vor, was jedoch beim Systemschaltungsentwurf automatisch durch die dort üblicherweise verwendete statische Timinganalyse berücksichtigt wird. Da diese Zeitdauer innerhalb der empfangenden Takt-Domäne vergeht, wird sie nicht der Synchronisationslatenz zugeschlagen.

[0039] Fig. 7 zeigt den Fall maximaler Latenz, wobei ein Zustandswechsel (von Low auf High, bzw. 0 auf 1, im vorliegenden Beispiel) des Eingangssignals iAsync am Anfang der Setup-Zeit des D-Flipflops L3 auftritt, d.h. zum Zeitpunkt $t_0{}' - t_{SETUP}$. Im ungünstigsten Fall kann das Flipflop L1 die Zustandsänderung des Eingangssignals iAsync nicht mehr

erfassen und das Flipflop L3 könnte die Zustandsänderung erfassen, wird aber metastabil und fällt nach Abklingen der Metastabilität auf den alten Wert zurück. Der Zeitpunkt $t_0' = t_0 + \Delta T$ bezeichnet den Zeitpunkt der gegenüber CLKa verzögerten Taktflanke des Taktsignals CLKb. In vorliegenden Beispiel nimmt das D-Flipflop L3 einen metastabilen Zustand ein und fällt wie erwähnt in den alten Zustand, und folglich kommt der aktuelle Zustand erst ungefähr zwei Perioden später, d.h. zum Zeitpunkt $t_2 = t_0 + 2$ in der Takt-Domäne von CLKa, im D-Flipflop L5 an. Die Synchronisationslatenz beträgt gemäß Fig. 7 $t_{SETUP} + (T_{CLK} - \Delta T) + T_{CLK} < 2T_{CLK}$. Die Signaländerung liegt synchron innerhalb der Takt-Domäne von CLKa um die Zeitdauer $t_{DEL} + t_F$ nach der steigenden Flanke von CLKa vor, was jedoch wiederum beim Systemschaltungsentwurf automatisch durch die dort üblicherweise verwendete statische Timinganalyse berücksichtigt wird. Da diese Zeitdauer innerhalb der empfangenden Takt-Domäne vergeht, wird sie nicht der Synchronisationslatenz zugeschlagen.

[0040] Fig. 8 illustriert eine Situation (Zustandsänderung im Eingangssignal iAsync zum Zeitpunkt $t_0 - t_{SETUP}$), die bei dem Beispiel aus Fig. 1 zur maximalen Latenz geführt hat. Bei der Schaltung gemäß Fig. 5 führt diese Situation zu einer signifikant kürzeren Latenz. Gemäß Fig. 8 wird in diesem Extremfall das Flipflop L1 metastabil und fällt dann auf den alten Wert zurück. Aufgrund des verzögerten Takts CLKb kann das Flipflop L3 den neuen Zustand des Eingangssignals zum Zeitpunkt t_0' korrekt erfassen. Bei der nächsten Taktflanke des Taktsignals CLKa manifestiert sich die Metastabilität in einer Inkonsistenz der Ausgangssignale A und B (d.h. A und B weisen verschiedene Werte auf). Diese Inkonsistenz wird von der kombinatorischen Logik CL1 (siehe Fig. 5) erkannt und innerhalb der Signallaufzeit t_F korrigiert (durch Invertierung des alten Wertes O). Das korrekte Ergebnis steht im Ausgangssignal Syncd=N bereits zum Zeitpunkt $t_1 + t_{DEL} + t_F$ zur Verfügung. Einen Takt später stellt das Flipflop L5 diesen neuen Wert als dann wiederum alten Wert O an seinem Ausgang zur Verfügung.

[0041] Der Durchsatz wird durch die kürzeste Zeitspanne zwischen zwei Signaländerungen bestimmt, bei der alle Signaländerungen in der empfangenden Takt-Domäne ankommen. Diese beträgt hier minimal T_{CLK}. Die empfangende Clock-Domain kann also die gleiche Taktfrequenz aufweisen, wie die (asynchron dazu) sendende Clock-Domain. Diese Zeitspanne wird erreicht, da stets höchstens eine der beiden Synchronisationsketten (Flipflops L1 + L2 und L3 + L4) ein Eingangs-Flipflop (Flipflop L1 oder L3) in einem met-

astabilen Zustand aufweist, während das jeweils andere Flipflop das Signal korrekt abtastet. In dem Beispiel aus Fig. 9 sind exemplarisch die Abtastzeitpunkte für unterschiedliche Phasenlagen des Eingangssignals iAsync gegenüber dem Takt CLKa durch Kreise hervorgehoben. Bei der ersten Variante des Eingangssignals iAsync (Fig. 9, drittes Diagramm von oben) tritt eine Änderung von 0 auf 1 vor dem Zeitpunkt $t_0 - t_{SETUP}$ auf und das Eingangssignal iAsync wird zum Zeitpunkt t_0 vom D-Flipflop L1 korrekt abgetastet. Bei der zweiten Variante des Eingangssignals iAsync (Fig. 9, viertes Diagramm von oben) tritt eine Änderung von 0 auf 1 nach dem Zeitpunkt $t_0 - t_{SETUP}$ jedoch vor dem Zeitpunkt t_0 auf, d.h. im D-Flipflop L1 kann Metastabilität auftreten und das D-Flipflop L3 ist das Sicher-Korrekt-Flipflop, welches zum Zeitpunkt $t_0' = t_0 + \Delta T$ das Eingangssignal iAsync korrekt abtastet. Das gleiche gilt für die dritte Variante des Eingangssignals iAsync (Fig. 9, fünftes Diagramm von oben), bei dem die Änderung von 0 auf 1 genau zum Zeitpunkt t_0 auftritt. Bei der vierten Variante des Eingangssignals iAsync (Fig. 9, sechstes Diagramm von oben) tritt eine Änderung von 0 auf 1 zum dem Zeitpunkt $t_0' - t_{SETUP}$ auf, und der alte Wert (also 0) des Eingangssignals iAsync wird zum Zeitpunkt t_0 vom D-Flipflop L1 abgetastet, wohingegen im D-Flipflop L3 Metastabilität auftreten kann. Das Gleiche gilt für die fünfte Variante des Eingangssignals iAsync (Fig. 9, siebtes Diagramm von oben), bei der die Änderung von 0 auf 1 im Eingangssignal iAsync zwischen $t_0' - t_{SETUP}$ und t_0' auftritt. Bei der sechsten Variante (Fig. 9, unterstes Diagramm) des Eingangssignals iAsync (Fig. 9, achtes Diagramm von oben) tritt die Änderung von 0 auf 1 zwischen t_0' und $t_0' + t_{HOLD}$ auf, und folglich tastet das D-Flipflop L1 noch den alten Wert ab und im D-Flipflop L3 kann Metastabilität auftreten. Bei der letzten Variante kommt die Änderung von 0 auf 1 im Eingangssignal iAsync erst nach dem Zeitpunkt $t_0' + t_{HOLD}$, d.h. das D-Flipflop L1 tastet zum Zeitpunkt t_0 noch den alten Wert ab und im D-Flipflop L3 kann Metastabilität auftreten. Erst zum Zeitpunkt t_1 kann das D-Flipflop L3 die Änderung von 0 auf 1 erfassen.

[0042] Die Zeitspanne zwischen zwei Änderungen des Eingangssignals iAsync darf in einem gewissen Rahmen die Zeitspanne T_{CLK} fallweise unterschreiten, wenn diese zumindest im zeitlichen Mittel eingehalten wird.

[0043] Es versteht sich, dass ein Fachmann in der Lage ist das Beispiel aus Fig. 5 zu modifizieren, ohne seine Funktion und dessen Wirkung nennenswert zu ändern. Beispielsweise können D-Flipflops auch durch andere Arten von flankengesteuerten Flipflops oder pegelgesteuerten Flipflops (Latches) implementiert werden. Des Weiteren ist es auch prob-

lemlos möglich, eines, mehrere oder alle der flankengesteuerten Flipflops mit den fallenden Flanken der Taktsignale zu triggern. Auch können die in Fig. 5 verwendeten Synchronisationsketten nach Fig. 1 durch Hintereinanderschaltung von mehr als zwei Flipflops zwecks Erhöhung der MTBF gebildet werden. Des Weiteren kann – abhängig von der in einer konkreten Anwendung verfügbaren Schaltungstechnologie – die Implementierung der einzelnen Schaltungskomponenten verschieden sein. Beispielsweise kann die in Fig. 5 durch eine Wahrheitstabelle definierte kombinatorische Logik CL1 auf viele verschiedene Arten durch Kombinationen verschiedener Gatter-Schaltungen implementiert werden. Des Weiteren versteht sich, dass abhängig von der Anwendung, ein binärer Wert „0“ oder ein logischer Zustand „falsch“ sowohl durch einen hohen Pegel als auch durch einen niedrigen Pegel des Eingangssignals iAsync repräsentiert werden kann. Entsprechendes gilt für den Wert „1“ bzw. den logischen Zustand „wahr“, der durch den jeweils inversen Pegel des Eingangssignals repräsentiert wird. Die Funktion der Schaltung wird auch nicht dadurch verändert, wenn sie insgesamt eine invertierende Charakteristik aufweist. Das heißt, ein Wert „1“ des Eingangssignals iAsync kann auch in einen Wert „0“ des Ausgangssignals Syncd übersetzt werden und umgekehrt.

PATENTANSPRÜCHE

1. Eine Schaltung, die aufweist:

eine erste Serienschaltung umfassend ein erstes Eingangs-Flipflop (L1) und ein erstes Ausgangs-Flipflop (L2) und eine zweite Serienschaltung umfassend ein zweites Eingangs-Flipflop (L3) und ein zweites Ausgangs-Flipflop (L4), wobei der ersten Serienschaltung und der zweiten Serienschaltung ein Eingangssignal (iAsync) zugeführt ist, dessen Pegel einen binären Wert repräsentiert; und

eine kombinatorische Logikschaltung (CL1), welche dazu ausgebildet ist, als Eingangswerte einen Ausgangswert (A) der ersten Serienschaltung, einen Ausgangswert (B) der zweiten Serienschaltung und einen gespeicherten alten Wert (O) eines synchronisierten Ausgangssignals (Syncd) zu empfangen und basierend darauf einen aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) zu erzeugen,

wobei das erste Eingangs-Flipflop (L1) mit einem ersten Taktsignal (CLKa) getaktet ist, und das zweite Eingangs-Flipflop (L3) mit einem zweiten Taktsignal (CLKb) getaktet ist, welches in Bezug auf das erste Taktsignal (CLKa) um eine Verzögerungszeit (ΔT) verzögert ist.

2. Die Schaltung gemäß Anspruch 1, die weiter aufweist;

ein weiteres Flipflop (L5), das dazu ausgebildet ist, den aktualisierten Wert (Syncd) zu speichern und in einem folgenden Taktzyklus als den alten Wert (O) der kombinatorische Logikschaltung (CL1) zuzuführen.

3. Die Schaltung gemäß Anspruch 1 oder 2,

wobei das erste Ausgangs-Flipflop (L2) und das zweites Ausgangs-Flipflop (L4) mit dem ersten Taktsignal (CLKa) getaktet sind.

4. Die Schaltung gemäß einem der Ansprüche 1 bis 3,

wobei das erste Eingangs-Flipflop (L1) und das zweite Eingangs-Flipflop (L3) eine Setup-Zeit (t_{SETUP}) und eine Hold-Zeit (t_{HOLD}) aufweisen und wobei die Verzögerungszeit (ΔT) größer ist als die Summe von Setup-Zeit (t_{SETUP}) und Hold-Zeit (t_{HOLD}).

5. Die Schaltung gemäß einem der Ansprüche 1 bis 4,

wobei die Logikschaltung (CL1) dazu ausgebildet ist, als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem Aus-

gangswert (A) der ersten Serienschaltung entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung gleich sind, und als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem negierten alten Wert (O) entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung nicht gleich sind.

6. Die Schaltung gemäß einem der Ansprüche 1 bis 5,

wobei die Logikschaltung (CL1) eine Übertragungscharakteristik aufweist, die durch die folgende Gleichung charakterisiert ist:

$$Syncd = (A \cdot B + A \cdot \neg O + B \cdot \neg O),$$

wobei Syncd den aktuellen Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet, A den Ausgangswert der ersten Serienschaltung und B den Ausgangswert der zweiten Serienschaltung bezeichnet, und O den gespeicherten, alten Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet.

7. Die Schaltung gemäß einem der Ansprüche 1 bis 6,

wobei die Eingangs-Flipflops (L1, L3) und die Ausgangs-Flipflops (L2, L4) taktflankengesteuerte D-Flipflops sind.

8. Ein Verfahren, das folgendes umfasst:

Abtasten eines Eingangssignals (iAsync) synchron zu einem ersten Taktsignal (CLKa) und Speichern des abgetasteten Wertes in einem ersten Eingangs-Flipflop (L1);

Abtasten des Eingangssignals (iAsync) synchron zu einem zweiten Taktsignal (CLKb) und Speichern des abgetasteten Wertes in einem zweiten Eingangs-Flipflop (L3), wobei das zweite Taktsignal (CLKb) in Bezug auf das erste Taktsignal (CLKa) um eine Verzögerungszeit (ΔT) verzögert ist;

Weiterreichen des in dem ersten Eingangs-Flipflop (L1) gespeicherten Wertes (MayMetaA) an ein erstes Ausgangs-Flipflop (L2) und Weiterreichen des in dem zweiten Eingangs-Flipflop (L3) gespeicherten Wertes (MayMetaB) an ein zweites Ausgangs-Flipflop (L4);

Verknüpfen der in dem ersten Ausgangs-Flipflop (L2) und dem zweiten Ausgangs-Flipflop (L4) gespeicherten Werte (A, B) mit einem gespeicherten alten Wert (O) eines synchronisierten Ausgangssignals (Syncd) mittels einer kombinatorischen Logik-

schaltung (CL1), wobei das Ergebnis der Verknüpfung den neuen Wert des synchronisierten Ausgangssignals (Syncd) repräsentiert.

9. Das Verfahren gemäß Anspruch 8,

Speichern des neuen Werts des synchronisierten Ausgangssignals (Syncd) in einem weiteren Flipflop (L5), das dazu ausgebildet ist, sodass dieser in einem folgenden Taktzyklus als gespeicherter alter Wert (O) zur Verfügung steht.

10. Das Verfahren gemäß Anspruch 8 oder 9,

wobei die Logikschaltung (CL1) dazu ausgebildet ist, als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem Ausgangswert (A) der ersten Serienschaltung entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung gleich sind, und als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem negierten alten Wert (O) entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung nicht gleich sind.

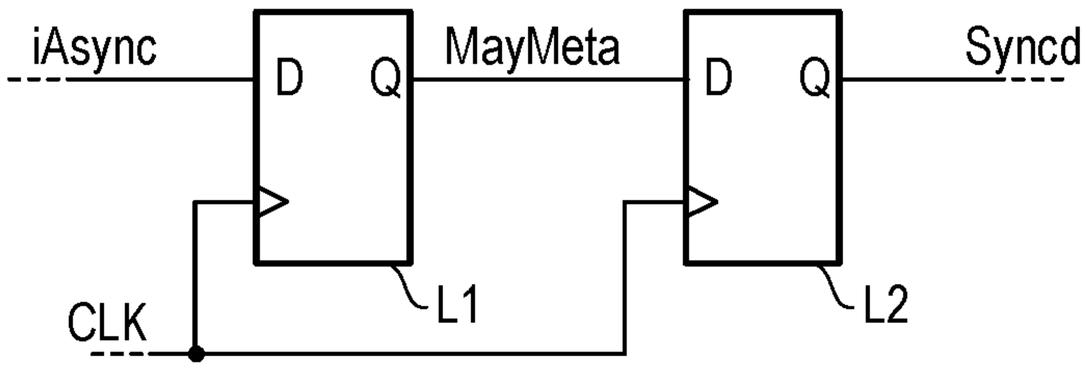

Fig. 1

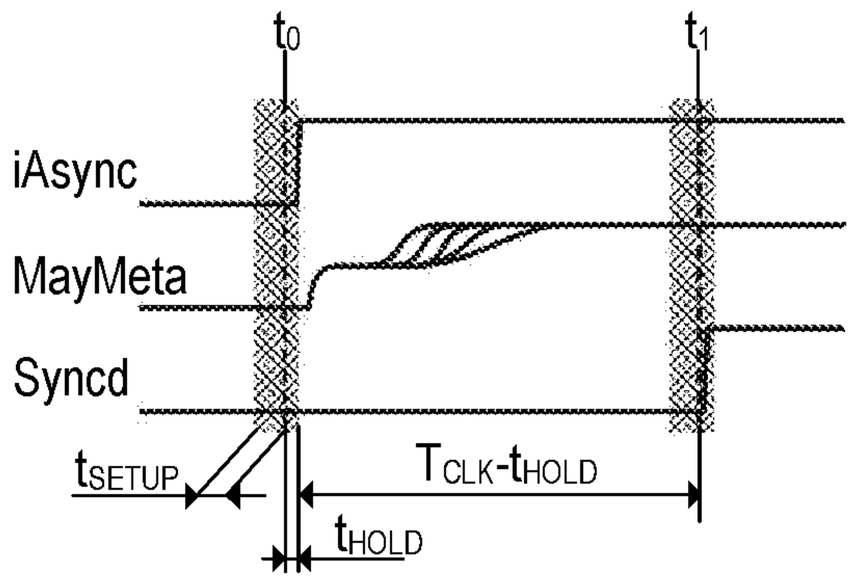

Fig. 2

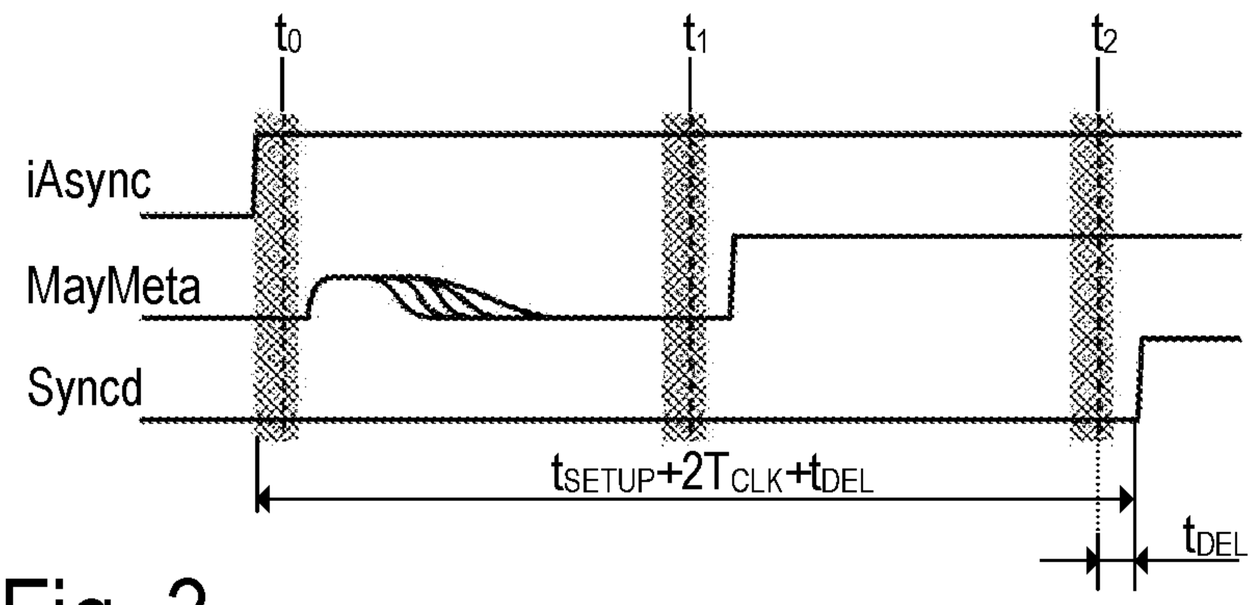

Fig. 3

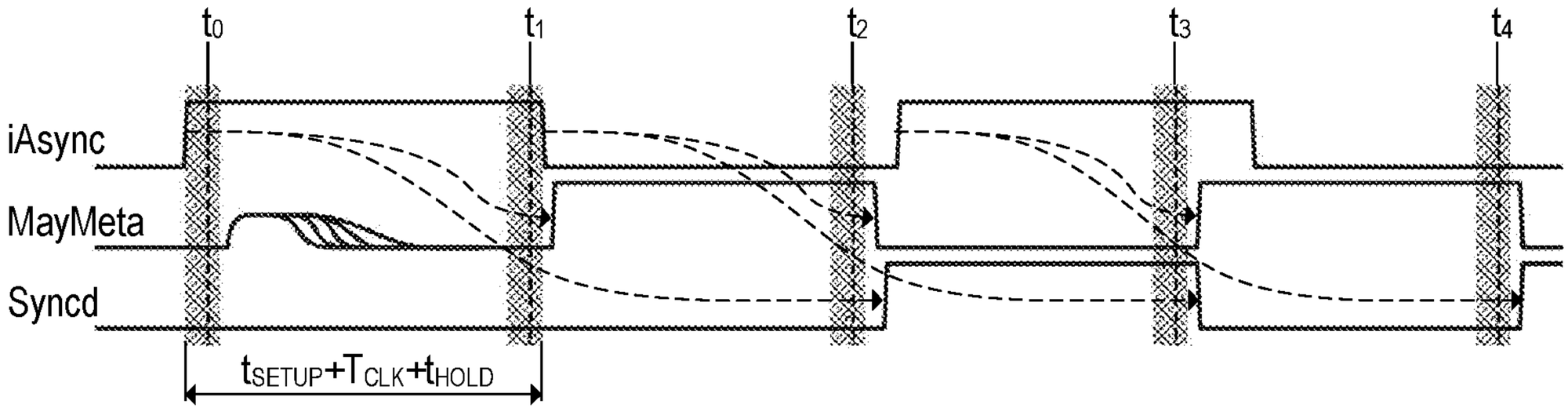

Fig. 4

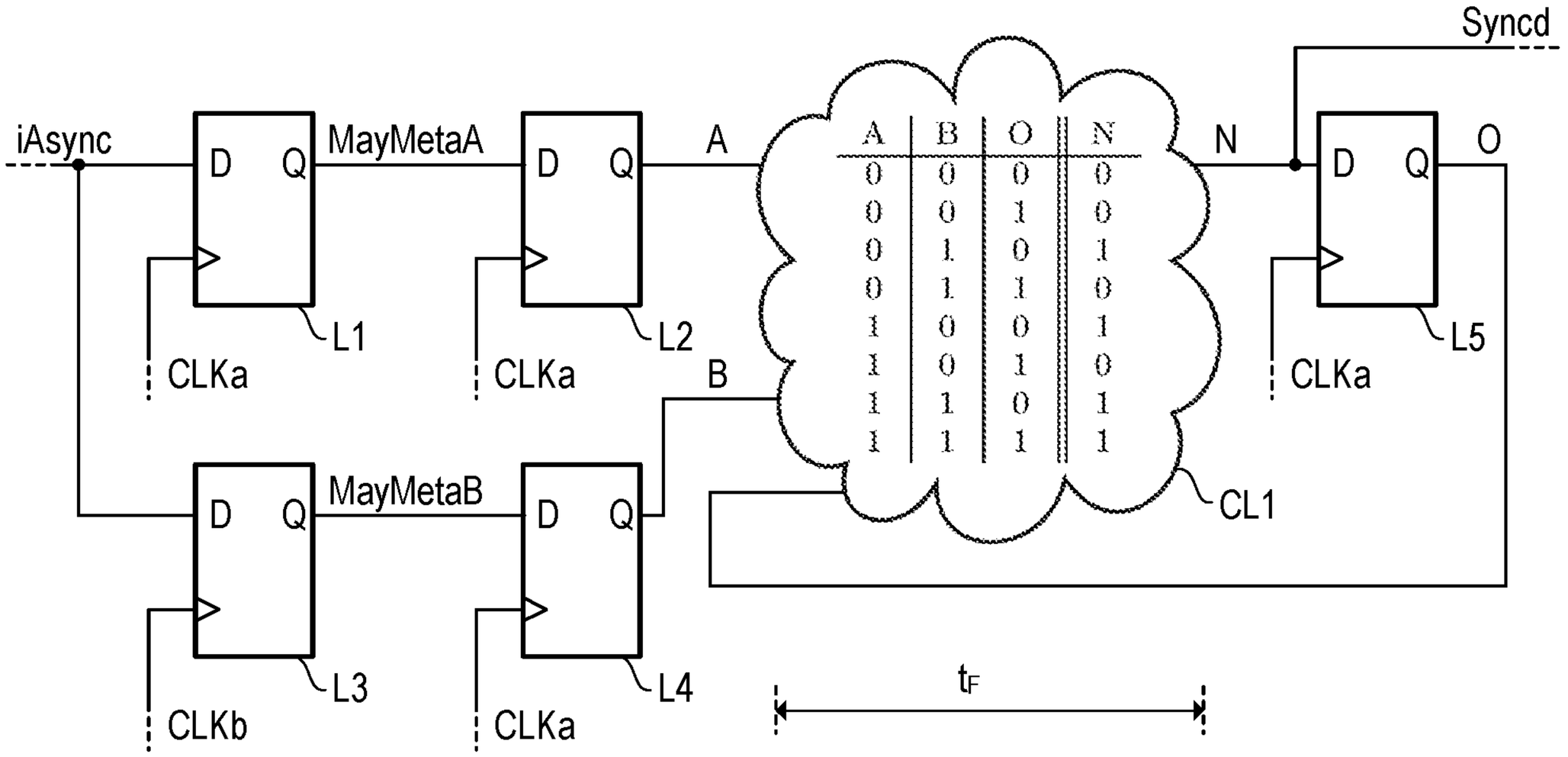
iAsync
D
Q
MayMetaA
D
Q
A
L1
CLKa
L2
CLKa
B
A
B
O
N
0 0 0 0
0 0 1 0
0 1 0 1
0 1 1 0
1 0 0 1
1 0 1 0
1 1 0 1
1 1 1 1
N
Syncd
D
Q
O
L5
CLKa
CL1
MayMetaB
D
Q
D
Q
L3
CLKb
L4
CLKa
t_F

Fig. 5

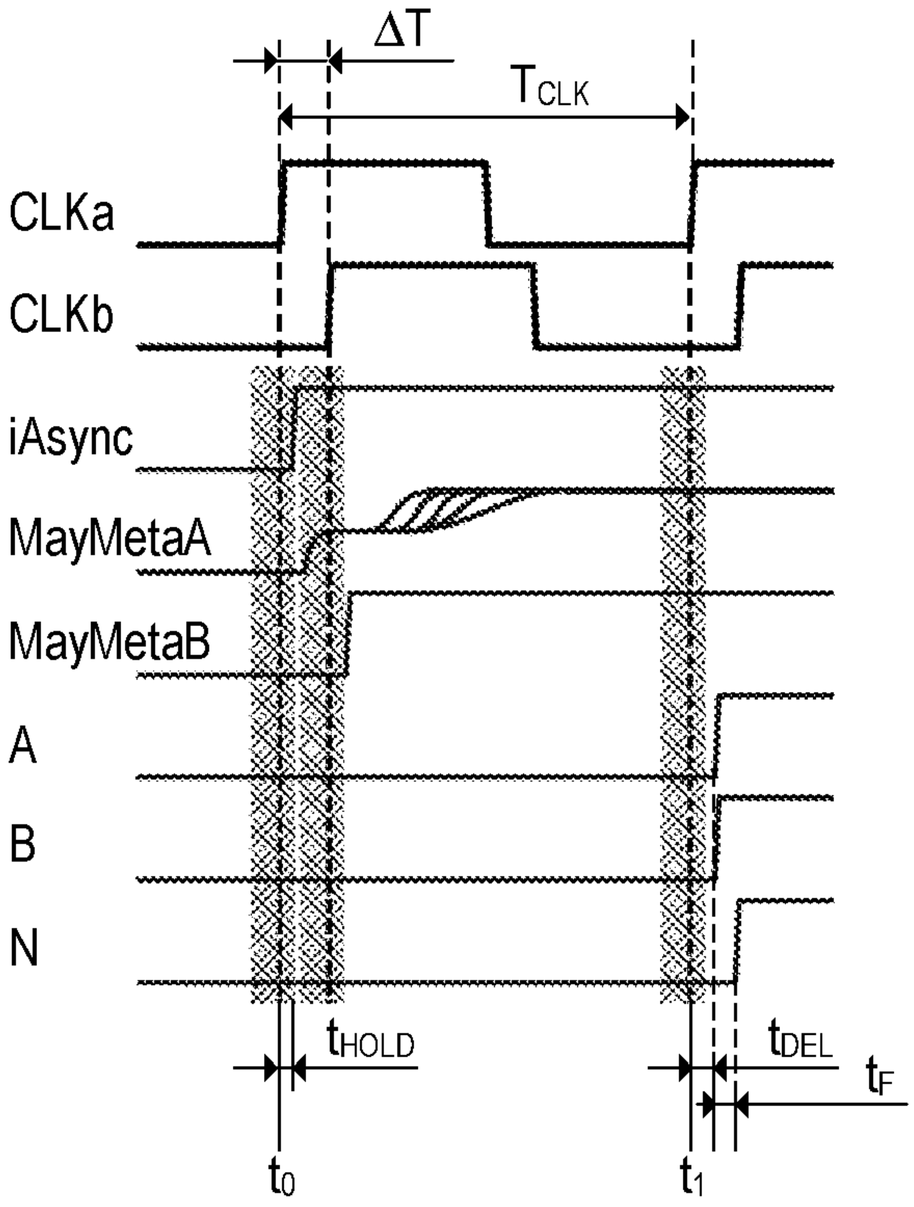
ΔT
T_{CLK}
CLKa
CLKb
iAsync
MayMetaA
MayMetaB
A
B
N
t_{HOLD}
t_{DEL}
t_F
t_0
t_1

Fig. 6

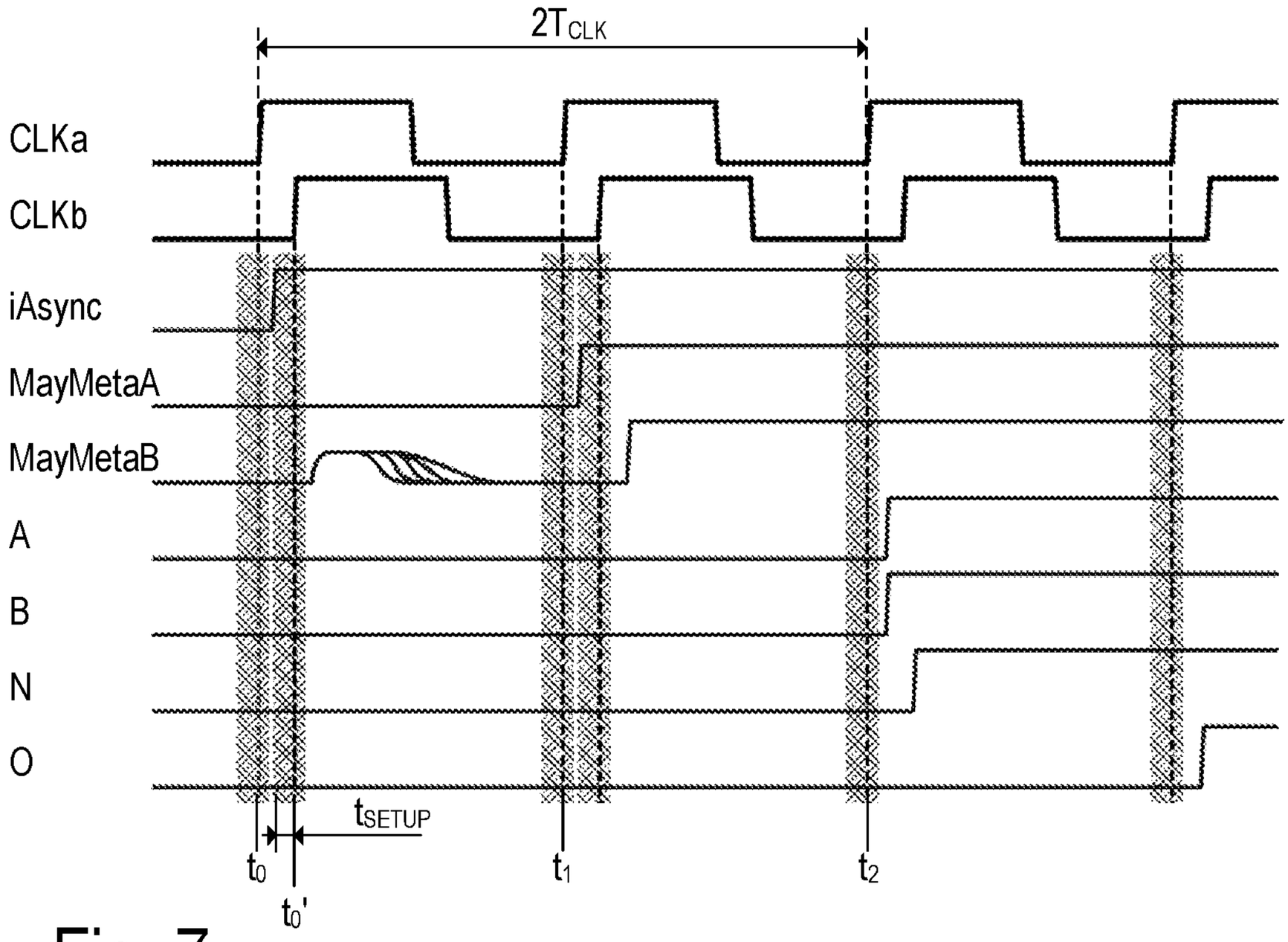
2T_CLK
CLKa
CLKb
iAsync
MayMetaA
MayMetaB
A
B
N
O
t_SETUP
t_0
t_0'
t_1
t_2

Fig. 7

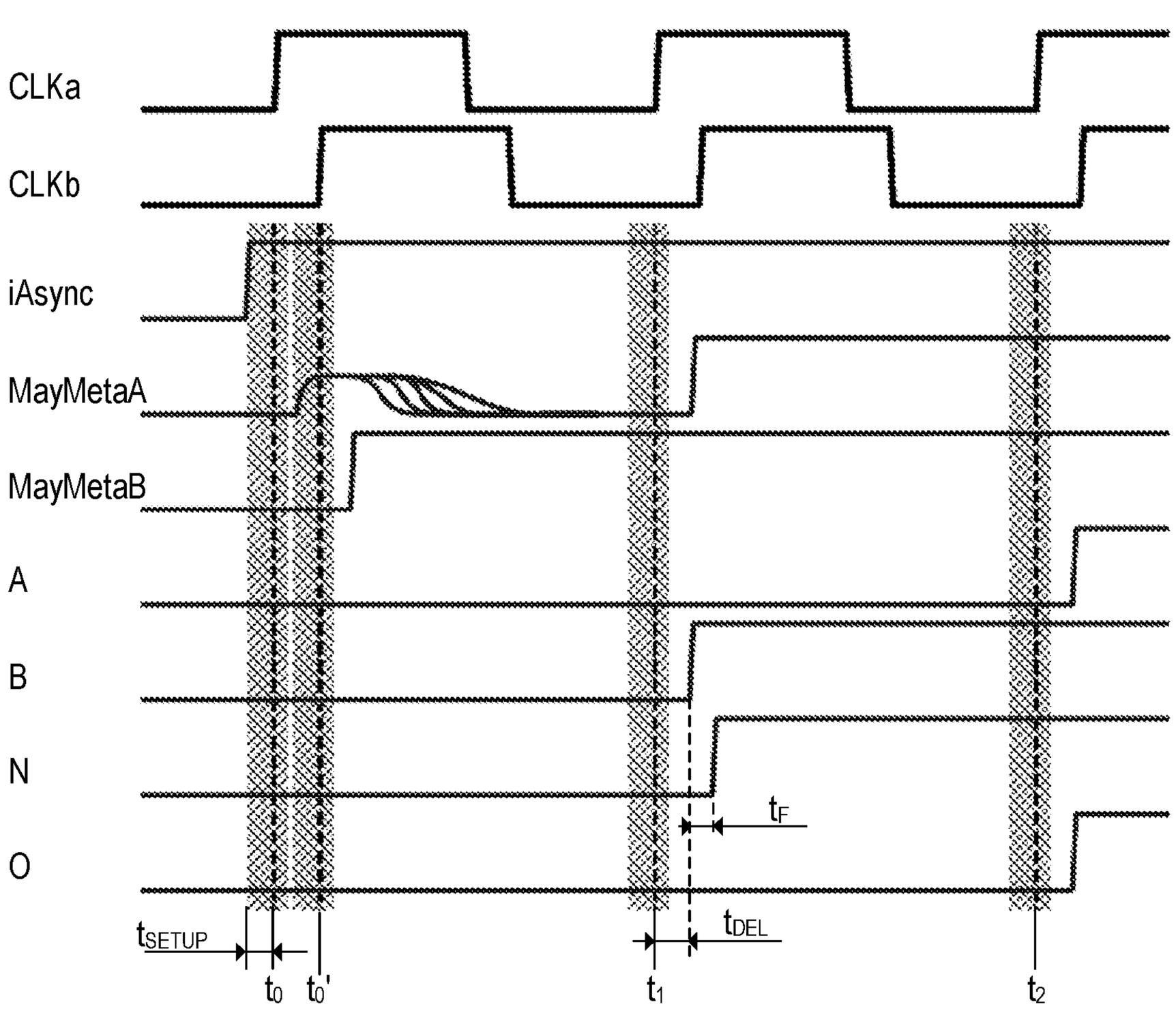
CLKa
CLKb
iAsync
MayMetaA
MayMetaB
A
B
N
O
t_F
t_SETUP
t_DEL
t_0
t_0'
t_1
t_2

Fig. 8

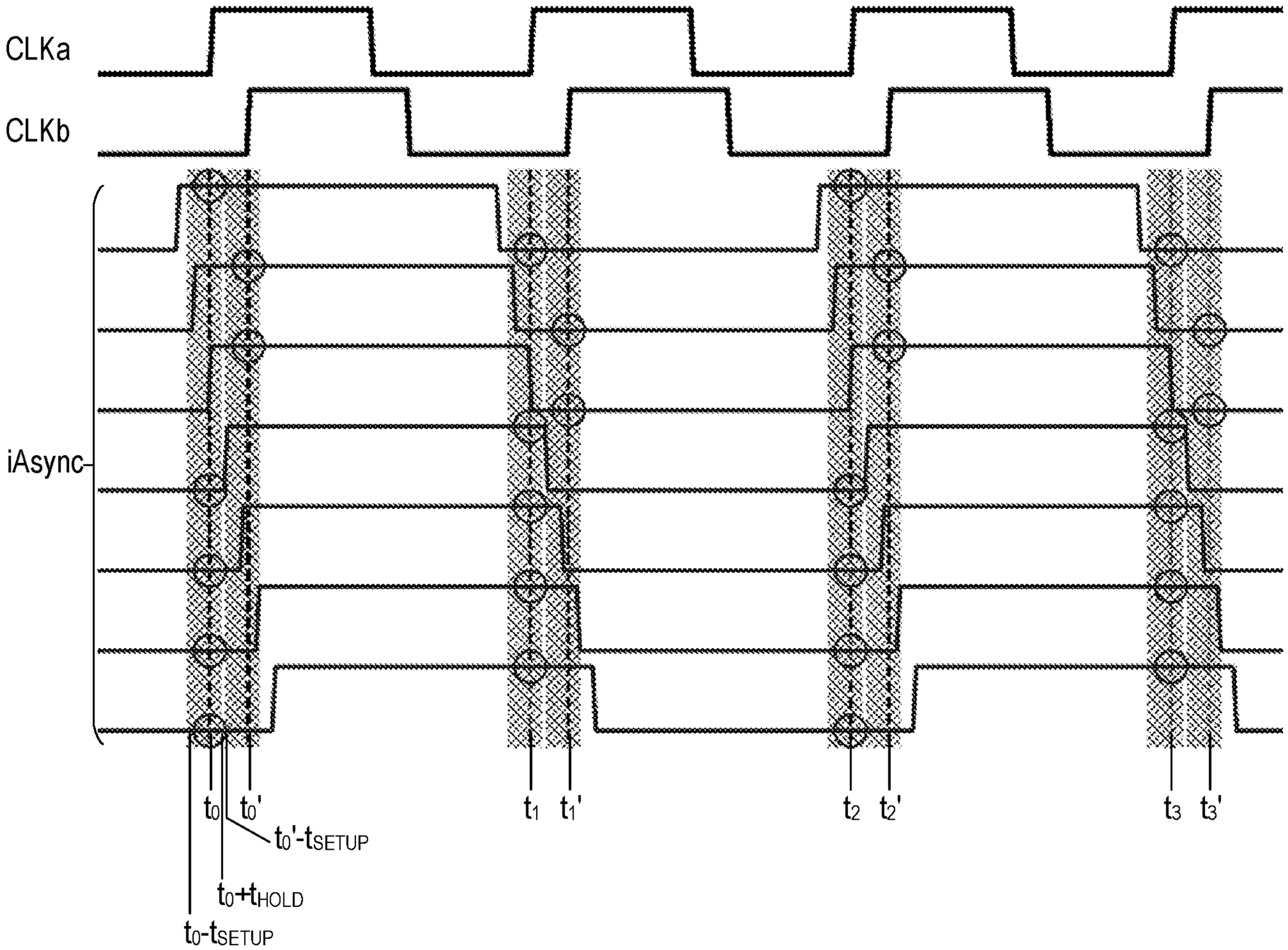
CLKa
CLKb
iAsync
t0
t0'
t0'-tSETUP
t0+tHOLD
t0-tSETUP
t1
t1'
t2
t2'
t3
t3'

Fig. 9

PATENTANSPRÜCHE

1. Eine Schaltung, die aufweist:

eine erste Serienschaltung umfassend ein erstes Eingangs-Flipflop (L1) und ein erstes Ausgangs-Flipflop (L2) und eine zweite Serienschaltung umfassend ein zweites Eingangs-Flipflop (L3) und ein zweites Ausgangs-Flipflop (L4), wobei der ersten Serienschaltung und der zweiten Serienschaltung ein Eingangssignal (iAsync) zugeführt ist, dessen Pegel einen binären Wert repräsentiert; und

eine kombinatorische Logikschaltung (CL1), welche dazu ausgebildet ist, als Eingangswerte einen Ausgangswert (A) der ersten Serienschaltung, einen Ausgangswert (B) der zweiten Serienschaltung und einen gespeicherten alten Wert (O) eines synchronisierten Ausgangssignals (Syncd) zu empfangen und basierend darauf einen aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) zu erzeugen,

wobei das erste Eingangs-Flipflop (L1) mit einem ersten Taktsignal (CLKa) getaktet ist, und das zweite Eingangs-Flipflop (L3) mit einem zweiten Taktsignal (CLKb) getaktet ist, welches in Bezug auf das erste Taktsignal (CLKa) um eine Verzögerungszeit (ΔT) verzögert ist, und

wobei die kombinatorische Logikschaltung (CL1) dazu ausgebildet ist, als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem Ausgangswert (A) der ersten Serienschaltung entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung gleich sind, und als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem negierten alten Wert (O) entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung nicht gleich sind.

2. Die Schaltung gemäß Anspruch 1, die weiter aufweist;

ein weiteres Flipflop (L5), das dazu ausgebildet ist, den aktualisierten Wert (Syncd) zu speichern und in einem folgenden Taktzyklus als den alten Wert (O) der kombinatorische Logikschaltung (CL1) zuzuführen.

3. Die Schaltung gemäß Anspruch 1 oder 2,

wobei das erste Ausgangs-Flipflop (L2) und das zweites Ausgangs-Flipflop (L4) mit dem ersten Taktsignal (CLKa) getaktet sind.

4. Die Schaltung gemäß einem der Ansprüche 1 bis 3,

ZULETZT VORGELEGTE ANSPRÜCHE

wobei das erste Eingangs-Flipflop (L1) und das zweite Eingangs-Flipflop (L3) eine Setup-Zeit (t_{SETUP}) und eine Hold-Zeit (t_{HOLD}) aufweisen und wobei die Verzögerungszeit (ΔT) größer ist als die Summe von Setup-Zeit (t_{SETUP}) und Hold-Zeit (t_{HOLD}).

5. Die Schaltung gemäß einem der Ansprüche 1 bis 4,

wobei die kombinatorische Logikschaltung (CL1) eine Übertragungscharakteristik aufweist, die durch die folgende Gleichung charakterisiert ist:

$$Syncd = (A \cdot B + A \cdot \neg O + B \cdot \neg O),$$

wobei Syncd den aktuellen Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet, A den Ausgangswert der ersten Serienschaltung und B den Ausgangswert der zweiten Serienschaltung bezeichnet, und O den gespeicherten, alten Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet.

6. Die Schaltung gemäß einem der Ansprüche 1 bis 5,

wobei die Eingangs-Flipflops (L1, L3) und die Ausgangs-Flipflops (L2, L4) taktflankengesteuerte D-Flipflops sind.

7. Ein Verfahren, das folgendes umfasst:

Abtasten eines Eingangssignals (iAsync) synchron zu einem ersten Taktsignal (CLKa) und Speichern des abgetasteten Wertes in einem ersten Eingangs-Flipflop (L1);

Abtasten des Eingangssignals (iAsync) synchron zu einem zweiten Taktsignal (CLKb) und Speichern des abgetasteten Wertes in einem zweiten Eingangs-Flipflop (L3), wobei das zweite Taktsignal (CLKb) in Bezug auf das erste Taktsignal (CLKa) um eine Verzögerungszeit (ΔT) verzögert ist;

Weiterreichen des in dem ersten Eingangs-Flipflop (L1) gespeicherten Wertes (MayMetaA) an ein erstes Ausgangs-Flipflop (L2) und Weiterreichen des in dem zweiten Eingangs-Flipflop (L3) gespeicherten Wertes (MayMetaB) an ein zweites Ausgangs-Flipflop (L4);

Verknüpfen der in dem ersten Ausgangs-Flipflop (L2) und dem zweiten Ausgangs-Flipflop (L4) gespeicherten Werte (A, B) mit einem gespeicherten alten Wert (O) eines synchronisierten Ausgangssignals (Syncd) mittels einer kombinatorischen Logik-

ZULETZT VORGELEGTE ANSPRÜCHE

schaltung (CL1), wobei das Ergebnis der Verknüpfung den neuen Wert des synchronisierten Ausgangssignals (Syncd) repräsentiert,

wobei die kombinatorische Logikschaltung (CL1) dazu ausgebildet ist, als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem Ausgangswert (A) der ersten Serienschaltung entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung gleich sind, und als aktualisierten Wert des synchronisierten Ausgangssignals (Syncd) einen Wert auszugeben, der dem negierten alten Wert (O) entspricht, wenn die Ausgangswerte (A, B) der ersten und der zweiten Serienschaltung nicht gleich sind.

8. Das Verfahren gemäß Anspruch 7,

Speichern des neuen Werts des synchronisierten Ausgangssignals (Syncd) in einem weiteren Flipflop (L5), das dazu ausgebildet ist, sodass dieser in einem folgenden Taktzyklus als gespeicherter alter Wert (O) zur Verfügung steht.

9. Das Verfahren gemäß Anspruch 7 oder 8,

wobei das erste Eingangs-Flipflop (L1) und das zweite Eingangs-Flipflop (L3) eine Setup-Zeit (t_{SETUP}) und eine Hold-Zeit (t_{HOLD}) aufweisen und wobei die Verzögerungszeit (ΔT) größer ist als die Summe von Setup-Zeit (t_{SETUP}) und Hold-Zeit (t_{HOLD}).

10. Das Verfahren gemäß einem der Ansprüche 7 bis 9,

wobei die kombinatorische Logikschaltung (CL1) eine Übertragungscharakteristik aufweist, die durch die folgende Gleichung charakterisiert ist:

$$Syncd = (A \cdot B + A \cdot \neg O + B \cdot \neg O),$$

wobei Syncd den aktuellen Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet, A den Ausgangswert der ersten Serienschaltung und B den Ausgangswert der zweiten Serienschaltung bezeichnet, und O den gespeicherten, alten Wert des synchronisierten Ausgangssignals (Syncd) bezeichnet.

ZULETZT VORGELEGTE ANSPRÜCHE