

(19)日本国特許庁(JP)

**(12)特許公報(B2)**

(11)特許番号  
**特許第7217153号**  
**(P7217153)**

(45)発行日 令和5年2月2日(2023.2.2)

(24)登録日 令和5年1月25日(2023.1.25)

(51)国際特許分類

G 0 4 G	5/00 (2013.01)	F I	G 0 4 G	5/00	J
H 0 4 Q	9/00 (2006.01)		H 0 4 Q	9/00	3 1 1 H
H 0 4 M	11/00 (2006.01)		H 0 4 M	11/00	3 0 1

請求項の数 6 (全28頁)

(21)出願番号 特願2019-2774(P2019-2774)  
 (22)出願日 平成31年1月10日(2019.1.10)  
 (65)公開番号 特開2020-112412(P2020-112412  
 A)  
 (43)公開日 令和2年7月27日(2020.7.27)  
 審査請求日 令和3年10月7日(2021.10.7)

(73)特許権者 000005223  
 富士通株式会社  
 神奈川県川崎市中原区上小田中4丁目1  
 番1号  
 (74)代理人 100121083  
 弁理士 青木 宏義  
 100138391  
 弁理士 天田 昌行  
 100074099  
 弁理士 大菅 義之  
 100133570  
 弁理士 德 永 民雄  
 清水 貴志  
 神奈川県川崎市中原区上小田中4丁目1  
 番1号 富士通株式会社内

最終頁に続く

(54)【発明の名称】 情報処理装置、時刻同期方法、及び時刻同期プログラム

**(57)【特許請求の範囲】****【請求項1】**

第1システム時刻を記憶する記憶部と、

前記記憶部に前記第1システム時刻が書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報を受信し、前記第1情報と前記第1時刻情報を受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報を受信する受信部と、

前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算する第1周波数計算部と、

前記受信部が前記第1情報と前記第1時刻情報を受信したときに、前記クロック信号に基づいて特定された時刻を、前記受信部が前記第1情報と前記第1時刻情報を受信した受信時刻として用い、前記第1システム時刻に基づいて、前記受信時刻を第2システム時刻に変換する受信時刻変換部と、

前記第2システム時刻と前記第1時刻情報を基づいて、前記第2時刻情報を第3システム時刻に変換する時刻情報変換部と、

前記第2システム時刻を前記第1情報に付加し、前記第3システム時刻を前記第2情報を付加する付加部と、

を備え、

前記時刻情報変換部は、

前記第2時刻情報と前記第1時刻情報との差分を生成する減算部と、  
前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算する第1差分計算部と、  
前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、前記第3システム時刻を求める第1加算部と、  
を含むことを特徴とする情報処理装置。

#### 【請求項2】

前記受信時刻変換部は、

前記受信時刻と前記クロック信号の周波数とを用いて、前記第2システム時刻と前記第1システム時刻との差分を計算する第2差分計算部と、

前記第2システム時刻と前記第1システム時刻との差分を前記第1システム時刻に加算することで、前記第2システム時刻を求める第2加算部と、  
 を含むことを特徴とする請求項1記載の情報処理装置。

#### 【請求項3】

前記第1システム時刻と前記クロック信号とを用いて、前記クロック信号の周波数を計算する第2周波数計算部をさらに備えることを特徴とする請求項2記載の情報処理装置。

#### 【請求項4】

前記受信部は、前記記憶部に前記第1システム時刻が書き込まれた後に、複数の情報取得装置のうちいずれかの情報取得装置から、前記いずれかの情報取得装置が取得した第1情報と第1時刻情報とを受信し、前記第1情報と前記第1時刻情報とを受信した後に、前記いずれかの情報取得装置から、第2情報と第2時刻情報とを受信し、

前記付加部は、前記第2システム時刻を前記いずれかの情報取得装置から受信した第1情報に付加し、前記第3システム時刻を前記いずれかの情報取得装置から受信した第2情報に付加することを特徴とする請求項1乃至3のいずれか1項に記載の情報処理装置。

#### 【請求項5】

情報処理装置によって実行される時刻同期方法であって、

前記情報処理装置が、

第1システム時刻が記憶部に書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報を受信し、

前記第1情報と前記第1時刻情報を受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報を受信し、

前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算し、

前記第1情報と前記第1時刻情報を受信したときに、前記クロック信号に基づいて特定された時刻を、前記第1情報と前記第1時刻情報を受信した受信時刻として用い、前記第1システム時刻に基づいて、前記受信時刻を第2システム時刻に変換し、

前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算し、

前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、第3システム時刻を求め、

前記第2システム時刻を前記第1情報に付加し、

前記第3システム時刻を前記第2情報に付加する、

ことを特徴とする時刻同期方法。

#### 【請求項6】

第1システム時刻が記憶部に書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報を受信し、

前記第1情報と前記第1時刻情報を受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報を受信し、

前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の

10

20

30

40

50

周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算し、

前記第1情報と前記第1時刻情報とを受信したときに、前記クロック信号に基づいて特定された時刻を、前記第1情報と前記第1時刻情報とを受信した受信時刻として用い、前記第1システム時刻に基づいて、前記受信時刻を第2システム時刻に変換し、

前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算し、

前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、第3システム時刻を求め、

前記第2システム時刻を前記第1情報に付加し、

前記第3システム時刻を前記第2情報に付加する、

処理をコンピュータに実行させるための時刻同期プログラム。

#### 【発明の詳細な説明】

##### 【技術分野】

##### 【0001】

本発明は、情報処理装置、時刻同期方法、及び時刻同期プログラムに関する。

##### 【背景技術】

##### 【0002】

図1は、複数のセンサから情報を収集する、従来の情報処理システムの構成例を示している。図1の情報処理システムは、情報処理装置101、センサ102-1、及びセンサ102-2を含む。情報処理装置101は、FPGA(Field-Programmable Gate Array)111を含む。

##### 【0003】

センサ102-1及びセンサ102-2は、個別のタイミング信号に同期して動作する。センサ102-1は、データ121-1～データ121-4を取得し、取得したデータをバッファリングしながら情報処理装置101へ送信する。同様に、センサ102-2は、データ122-1～データ122-4を取得し、取得したデータをバッファリングしながら情報処理装置101へ送信する。

##### 【0004】

FPGA111は、センサ102-1からデータ121-1～データ121-4を受信し、センサ102-2からデータ122-1～データ122-4を受信する。そして、FPGA111は、受信したデータに対する情報処理を行う。

##### 【0005】

センサに関連して、複数のセンサによる計測結果同士を時間的に対応付けるセンサ情報処理装置、及びセンサデータに関する誤差の少ないタイムスタンプが得られるセンシングシステムが知られている（例えば、特許文献1及び特許文献2を参照）。

##### 【先行技術文献】

##### 【特許文献】

##### 【0006】

##### 【文献】特開2015-19223号公報

国際公開第2018/151202号パンフレット

##### 【発明の概要】

##### 【発明が解決しようとする課題】

##### 【0007】

情報処理装置において複数のセンサから収集した情報を処理する際に、各情報に時刻を示すタイムスタンプを付加する場合がある。しかし、各センサが個別のタイミング信号に同期して動作している場合、複数のセンサのタイミング信号に基づく時刻は互いに同期しておらず、情報処理装置のシステム時刻とも同期していない。

##### 【0008】

なお、時刻同期の問題は、複数のセンサから情報を収集する場合に限らず、単一のセン

サから情報を収集する場合においても生ずるものである。

**【0009】**

1つの側面において、本発明は、情報取得装置が情報を取得した時刻を、取得された情報を処理する情報処理装置のシステム時刻と同期させることを目的とする。

**【課題を解決するための手段】**

**【0010】**

1つの案では、情報処理装置は、記憶部、受信部、受信時刻変換部、時刻情報変換部、及び付加部を含む。

**【0011】**

記憶部は、第1システム時刻を記憶する。受信部は、記憶部に第1システム時刻が書き込まれた後に、情報取得装置から、情報取得装置が取得した第1情報と、第1情報の取得時刻を示す第1時刻情報を受信する。そして、受信部は、第1情報と第1時刻情報を受信した後に、情報取得装置から、第2情報と第2情報の取得時刻を示す第2時刻情報を受信する。

10

**【0012】**

受信時刻変換部は、第1システム時刻に基づいて、受信部が第1情報と第1時刻情報を受信した受信時刻を、第2システム時刻に変換し、時刻情報変換部は、第2システム時刻と第1時刻情報を基づいて、第2時刻情報を第3システム時刻に変換する。付加部は、第2システム時刻を第1情報に付加し、第3システム時刻を第2情報に付加する。

**【発明の効果】**

20

**【0013】**

実施形態によれば、情報取得装置が情報を取得した時刻を、取得された情報を処理する情報処理装置のシステム時刻と同期させることができる。

**【図面の簡単な説明】**

**【0014】**

**【図1】**従来の情報処理システムの構成図である。

**【図2】**FPGAの第1の比較例を示す図である。

**【図3】**FPGAの第2の比較例を示す図である。

**【図4】**情報処理装置の機能的構成図である。

**【図5】**時刻同期処理のフローチャートである。

30

**【図6】**実施形態の情報処理システムの構成図である。

**【図7】**センサモジュールの構成図である。

**【図8】**ビデオモジュールの構成図である。

**【図9】**変換部の構成図である。

**【図10A】**時刻同期処理の具体例を示すフローチャート(その1)である。

**【図10B】**時刻同期処理の具体例を示すフローチャート(その2)である。

**【図11】**時刻同期処理を示す図である。

**【図12】**第2周波数計算部の構成図である。

**【図13】**第1周波数計算部の構成図である。

**【図14】**クロック信号の周波数を示す図である。

40

**【図15】**タイミング信号の周波数を示す図である。

**【図16】**OS時刻を示す図である。

**【図17】**パケット消失時の時刻同期処理を示す図である。

**【図18】**ソフトウェア制御による時刻同期部の構成図である。

**【発明を実施するための形態】**

**【0015】**

以下、図面を参照しながら、実施形態を詳細に説明する。

図2は、データにタイムスタンプを付加するFPGAの第1の比較例を示している。図2のFPGA201は、割り込みコントローラ211、MMIO(Memory-mapped Input Output)レジスタ212、及び検出部213を含む。

50

**【 0 0 1 6 】**

検出部 213 は、センサから入力データを受信すると、割り込みコントローラ 211 へ検出信号を出力する。割り込みコントローラ 211 は、検出信号を受信すると、オペレーティングシステム (Operating System, OS) 202 に対して、割り込み又はポーリングにより時刻同期を要求する。OS 202 は、割り込みコントローラ 211 からの要求に基づいて OS 時刻 221 を取得し、FPGA 201 のMMIOL レジスタ 212 に書き込む。OS 時刻 221 は、システム時刻の一例である。

**【 0 0 1 7 】**

FPGA 201 は、MMIOL レジスタ 212 に書き込まれた OS 時刻 221 を、タイムスタンプとして、検出部 213 から出力される入力データに付加し、タイムスタンプが付加された出力データを出力する。10

**【 0 0 1 8 】**

しかしながら、図 2 の構成では、出力データのタイムスタンプは、FPGA 201 が入力データを受信したときの OS 時刻を示しており、センサが入力データを取得したときの OS 時刻を示していない。

**【 0 0 1 9 】**

図 3 は、データにタイムスタンプを付加する FPGA の第 2 の比較例を示している。図 3 の FPGA 301 は、MMIOL レジスタ 311 及び加算部 312 を含む。

**【 0 0 2 0 】**

センサは、タイミング信号に同期してカウント値をインクリメントするカウンタを含み、入力データを取得したときのカウント値をその入力データに付加して、FPGA 301 へ送信する。20

**【 0 0 2 1 】**

OS 202 は、定期的に OS 時刻 221 を取得し、FPGA 301 のMMIOL レジスタ 311 に書き込む。FPGA 301 は、MMIOL レジスタ 311 に OS 時刻 221 が書き込まれると、リセット信号をセンサへ送信し、センサは、リセット信号を受信すると、カウンタをリセットする。

**【 0 0 2 2 】**

FPGA 301 は、センサから入力データを受信すると、入力データからカウント値を取得する。加算部 312 は、MMIOL レジスタ 311 に書き込まれた OS 時刻 221 に、カウント値が示す時間を加算することで、タイムスタンプを生成する。そして、FPGA 301 は、生成されたタイムスタンプを入力データに付加し、タイムスタンプが付加された出力データを出力する。30

**【 0 0 2 3 】**

しかしながら、図 3 の構成では、入力データに付加されたカウント値に、FPGA 301 からセンサへリセット信号を送信する通信のオーバヘッドが含まれている。このため、そのカウント値から生成されるタイムスタンプは、必ずしも、センサが入力データを取得したときの OS 時刻を示していない。

**【 0 0 2 4 】**

図 4 は、実施形態の情報処理装置の機能的構成例を示している。図 4 の情報処理装置 401 は、記憶部 411、受信部 412、受信時刻変換部 413、時刻情報変換部 414、及び付加部 415 を含む。記憶部 411 は、第 1 システム時刻を記憶する。40

**【 0 0 2 5 】**

図 5 は、図 4 の情報処理装置 401 が行う時刻同期処理の例を示すフローチャートである。まず、受信部 412 は、記憶部 411 に第 1 システム時刻が書き込まれた後に、情報取得装置 402 から、情報取得装置 402 が取得した第 1 情報と、第 1 情報の取得時刻を示す第 1 時刻情報とを受信する (ステップ 501)。そして、受信部 412 は、第 1 情報と第 1 時刻情報とを受信した後に、情報取得装置 402 から、第 2 情報と第 2 情報の取得時刻を示す第 2 時刻情報とを受信する (ステップ 502)。

**【 0 0 2 6 】**

受信時刻変換部 413 は、第 1 システム時刻に基づいて、受信部 412 が第 1 情報と第 1 時刻情報とを受信した受信時刻を、第 2 システム時刻に変換する（ステップ 503）。時刻情報変換部 414 は、第 2 システム時刻と第 1 時刻情報に基づいて、第 2 時刻情報を第 3 システム時刻に変換する（ステップ 504）。そして、付加部 415 は、第 2 システム時刻を第 1 情報に付加し（ステップ 505）、第 3 システム時刻を第 2 情報に付加する（ステップ 506）。

#### 【0027】

図 4 の情報処理装置 401 によれば、情報取得装置が情報を取得した時刻を、取得された情報を処理する情報処理装置のシステム時刻と同期させることができる。

#### 【0028】

図 6 は、実施形態の情報処理システムの構成例を示している。図 6 の情報処理システムは、情報処理装置 601 と情報取得装置 602-1 ~ 情報取得装置 602-N (N は 1 以上の整数) を含む。情報処理装置 601 及び各情報取得装置 602-i (i = 1 ~ N) は、図 4 の情報処理装置 401 及び情報取得装置 402 にそれぞれ対応する。情報処理装置 601 と各情報取得装置 602-i は、通信ネットワーク 603 を介して通信することができる。

#### 【0029】

情報取得装置 602-i は、各種センサ、撮像装置、マイクロフォン等を含み、所定の情報を取得する。例えば、情報取得装置 602-i が圧力センサを含む場合、気圧、水圧等を示す情報が取得され、情報取得装置 602-i が温度センサを含む場合、温度を示す情報が取得される。情報取得装置 602-i が撮像装置を含む場合、画像情報が取得され、情報取得装置 602-i がマイクロフォンを含む場合、音声情報が取得される。情報取得装置 602-i に含まれるセンサは、湿度センサ、風力センサ等であってもよい。

#### 【0030】

情報取得装置 602-i は、取得した情報に、その情報の取得時刻を示す時刻情報を附加することで、センサデータのパケットを生成し、生成したパケットを情報処理装置 601 へ送信する。情報取得装置 602-i は、タイミング信号を用いて時刻情報を生成する。

#### 【0031】

情報処理装置 601 は、CPU (Central Processing Unit) 611、メモリ 612、及び時刻同期部 613 を含む。CPU 611 (プロセッサ) は、NTP (Network Time Protocol) によって定期的に更新される OS 時刻を、時刻同期部 613 へ出力する。

#### 【0032】

時刻同期部 613 は、CPU 611 が output する OS 時刻に基づいて、情報取得装置 602-i から受信したパケットに含まれる所定の情報に、その情報の取得時刻を示す OS 時刻を付加することで、転送データを生成する。そして、時刻同期部 613 は、生成した転送データをメモリ 612 へ転送し、メモリ 612 は、転送された転送データを記憶する。CPU 611 は、メモリ 612 が記憶している転送データに対する情報処理を行う。

#### 【0033】

時刻同期部 613 は、受信部 621、第 1 周波数計算部 622-1 ~ 第 1 周波数計算部 622-N、第 1 周波数計算部 623、変換部 624-1 ~ 変換部 624-N、及び付加部 625 を含む。受信部 621 及び付加部 625 は、図 4 の受信部 412 及び付加部 415 にそれぞれ対応する。

#### 【0034】

受信部 621 は、情報取得装置 602-i からパケットを受信し、パケットに含まれる所定の情報を付加部 625 へ出力し、パケットに含まれる時刻情報を、第 1 周波数計算部 622-i 及び変換部 624-i へ出力する。

#### 【0035】

第 2 周波数計算部 623 は、CPU 611 が output する OS 時刻と、時刻同期部 613 内のクロック信号とを用いて、そのクロック信号の周波数を計算する。そして、第 2 周波数計算部 623 は、計算したクロック信号の周波数を、第 1 周波数計算部 622-1 ~ 第 1

10

20

30

40

50

周波数計算部 622-N 及び変換部 624-1 ~ 変換部 624-N へ出力する。

#### 【0036】

第1周波数計算部 622-i は、2つのパケットに含まれる時刻情報の差分と、時刻同期部 613 内のクロック信号と、第2周波数計算部 623 が output する、クロック信号の周波数とを用いて、情報取得装置 602-i 内のタイミング信号の周波数を計算する。そして、第1周波数計算部 622-i は、計算したタイミング信号の周波数を変換部 624-i へ出力する。

#### 【0037】

変換部 624-i は、CPU 611 が output する OS 時刻に基づいて、受信部 412 が出力する時刻情報を OS 時刻に変換し、その OS 時刻を付加部 625 へ出力する。付加部 625 は、変換部 624-i が output する OS 時刻を、受信部 412 が出力する所定の情報にタイムスタンプとして付加することで、転送データを生成し、メモリ 612 へ転送する。

10

#### 【0038】

図7は、情報取得装置 602-i の一例であるセンサモジュールの構成例を示している。図7のセンサモジュール 701 は、圧力センサ 711、水晶振動子 712、AD (Analogue-to-Digital) コンバータ 713、カウンタ 714、書き込み回路 715、データバッファ 716、及び I2C (Inter-Integrated Circuit) バスマスター 717 を含む。この場合、図6の通信ネットワーク 603 として I2C バスが用いられる。

#### 【0039】

圧力センサ 711 は、圧力を示すアナログ信号を AD コンバータ 713 へ出力し、水晶振動子 712 は、正弦波のタイミング信号を AD コンバータ 713、カウンタ 714、及び I2C バスマスター 717 へ出力する。

20

#### 【0040】

カウンタ 714 は、水晶振動子 712 が output するタイミング信号に同期してカウント動作を行い、カウント値を書き込み回路 715 へ出力する。カウント値は、情報の取得時刻を示す時刻情報の一例である。AD コンバータ 713 は、水晶振動子 712 が output するタイミング信号に同期して、圧力センサ 711 が output するアナログ信号をデジタル信号の情報 721 に変換し、書き込み回路 715 へ出力する。

#### 【0041】

書き込み回路 715 は、AD コンバータ 713 が output する情報 721 に、カウンタ 714 が output するカウント値を付加することで、センサデータのパケットを生成して、データバッファ 716 へ出力する。データバッファ 716 は、複数のパケットをバッファリングして、I2C バスマスター 717 へ出力する。

30

#### 【0042】

I2C バスマスター 717 は、水晶振動子 712 が output するタイミング信号に同期して動作し、データバッファ 716 が output するパケットを、シリアルデータ SDA として情報処理装置 601 へ送信する。I2C バスマスター 717 は、シリアルデータ SDA とともに、シリアルクロック SCL を情報処理装置 601 へ送信する。

#### 【0043】

図8は、情報取得装置 602-i の一例であるビデオモジュールの構成例を示している。図8のビデオモジュール 801 は、撮像装置 811、水晶振動子 812、AD コンバータ 813、カウンタ 814、書き込み回路 815、及びデータバッファ 816 を含む。

40

#### 【0044】

撮像装置 811 は、画像センサを含み、デジタル映像を撮影して、VGA (Video Graphics Array) 信号を AD コンバータ 813 へ出力する。VGA 信号は、Hsync (Horizontal synchronizing signal)、Vsync (Vertical synchronizing signal)、blue ビデオ信号、green ビデオ信号、及び red ビデオ信号を含む。Hsync は、書き込み回路 815 にも出力される。水晶振動子 812 は、正弦波のタイミング信号をカウンタ 814 へ出力する。

#### 【0045】

50

カウンタ 814 は、水晶振動子 812 が出力するタイミング信号に同期してカウント動作を行い、カウント値を書き込み回路 815 へ出力する。AD コンバータ 813 は、撮像装置 811 が出力する VGA 信号をデジタル信号の画像情報 821 に変換し、書き込み回路 815 へ出力する。AD コンバータ 813 の代わりに、デジタルエンコーダを用いてよい。

#### 【0046】

書き込み回路 815 は、Hsync に同期して動作し、AD コンバータ 813 が出力する画像情報 821 に、カウンタ 814 が出力するカウント値を付加することで、センサデータのパケットを生成して、データバッファ 816 へ出力する。データバッファ 816 は、複数のパケットをバッファリングして、情報処理装置 601 へ送信する。

10

#### 【0047】

図 6 の時刻同期部 613 は、ハードウェア制御又はソフトウェア制御によって実現される。ハードウェア制御による時刻同期部 613 としては、FPGA のようなハードウェア論理回路が用いられ、ソフトウェア制御による時刻同期部 613 としては、コンピュータが用いられる。

#### 【0048】

以下では、図 9～図 17 を参照しながら、時刻同期部 613 として FPGA を用いた場合の構成及び動作を説明する。

#### 【0049】

図 9 は、図 6 の変換部 624-i の構成例を示している。図 9 の変換部 624-i は、書き込み回路 901、判定回路 902、セレクタ 903、カウンタ 904、レジスタ 905～レジスタ 911、及び減算器 (sub) 912 を含む。変換部 624-i は、さらに、DSP (Digital Signal Processor) 913、DSP 914、加算器 (add) 915 及び加算器 916 を含む。

20

#### 【0050】

レジスタ 905 は、図 4 の記憶部 411 に対応し、DSP 913 及び加算器 915 は、受信時刻変換部 413 に対応し、減算器 912、DSP 914、及び加算器 916 は、時刻情報変換部 414 に対応する。減算器 912 は減算部の一例であり、DSP 914 は第 1 差分計算部の一例であり、加算器 916 は第 1 加算部の一例である。DSP 913 は第 2 差分計算部の一例であり、加算器 915 は第 2 加算部の一例である。

30

#### 【0051】

書き込み回路 901 は、CPU 611 が出力する OS 時刻である OS\_system\_time を受信すると、OS\_system\_time を OS\_time としてレジスタ 905 に書き込む。レジスタ 905 は、OS\_time を加算器 915 へ出力する。

#### 【0052】

そして、書き込み回路 901 は、カウンタ 904 をクリアして、カウント値 FPGA\_cnt に 0 を設定し、レジスタ 909 が記憶するフラグ first\_pkt\_flag をクリアして、first\_pkt\_flag に論理 “0” を設定する。その後、カウンタ 904 は、時刻同期部 613 内のクロック信号 FPGA\_clock に同期してカウント動作を行い、カウント値 FPGA\_cnt をレジスタ 906 へ出力する。

40

#### 【0053】

受信部 621 は、情報取得装置 602-i からパケットを受信したとき、パケットの受信を示す信号 sensor\_valid と、パケットに含まれるカウント値 sensor\_count とを、第 1 周波数計算部 622-i 及び変換部 624-i へ出力する。sensor\_valid 及び sensor\_count は、判定回路 902 及びセレクタ 903 にそれぞれ入力される。

#### 【0054】

判定回路 902 は、sensor\_valid を受信した場合、レジスタ 909 の first\_pkt\_flag をチェックする。first\_pkt\_flag が論理 “0” である場合、判定回路 902 は、first\_pkt\_flag に論理 “1” を設定し、カウンタ 9

50

04が出力するFPGA\_cntを、1st\_pk\_t\_ckとしてレジスタ906に書き込む。

#### 【0055】

したがって、OS\_system\_timeがレジスタ905に書き込まれた後、受信部621が最初のパケットを受信したときに、first\_pk\_tf1agに論理“1”が設定され、そのときのFPGA\_cntが1st\_pk\_t\_ckに設定される。1st\_pk\_t\_ckは、最初のパケットを受信した受信時刻を示している。レジスタ906は、1st\_pk\_t\_ckをDSP913へ出力する。

#### 【0056】

次に、判定回路902は、セレクタ903にレジスタ907を選択させ、セレクタ903は、sensor\_cntをレジスタ907へ出力し、レジスタ907は、sensor\_cntを1st\_pk\_t\_cntとして記憶する。そして、判定回路902は、セレクタ903にレジスタ908を選択させ、セレクタ903は、sensor\_cntをレジスタ908へ出力し、レジスタ908は、sensor\_cntをcurr\_pk\_t\_cntとして記憶する。

#### 【0057】

したがって、受信部621が最初のパケットを受信したとき、そのパケットのsensor\_cntが、1st\_pk\_t\_cnt及びcurr\_pk\_t\_cntに設定される。

#### 【0058】

一方、first\_pk\_tf1agが論理“1”である場合、判定回路902は、セレクタ903にレジスタ908を選択させ、レジスタ908は、sensor\_cntをcurr\_pk\_t\_cntとして記憶する。この場合、first\_pk\_tf1ag、1st\_pk\_t\_ck、及び1st\_pk\_t\_cntは更新されない。

#### 【0059】

したがって、OS\_system\_timeがレジスタ905に書き込まれた後、受信部621が2番目以降のパケットを受信したとき、そのパケットのsensor\_cntがcurr\_pk\_t\_cntに設定され、1st\_pk\_t\_ck及び1st\_pk\_t\_cntは変化しない。

#### 【0060】

このように、1st\_pk\_t\_cntは、最初のパケットのsensor\_cntを表し、curr\_pk\_t\_cntは、最初のパケット又は2番目以降のパケットのsensor\_cntを表す。レジスタ907及びレジスタ908は、1st\_pk\_t\_cnt及びcurr\_pk\_t\_cntをそれぞれ減算器912へ出力する。減算器912は、curr\_pk\_t\_cntから1st\_pk\_t\_cntを減算することで、カウント値の差分を求め、DSP914へ出力する。

#### 【0061】

したがって、受信部621が最初のパケットを受信したとき、減算器912は、カウント値の差分として0を出力する。一方、受信部621が2番目以降のパケットを受信したとき、減算器912は、受信したパケットのsensor\_cntと最初のパケットのsensor\_cntとの差分を出力する。

#### 【0062】

第2周波数計算部623は、クロック信号FPGA\_ckの周波数FPGA\_freqを、第1周波数計算部622-1～第1周波数計算部622-N及び変換部624-1～変換部624-Nへ出力する。レジスタ910は、FPGA\_freqを記憶し、DSP913へ出力する。

#### 【0063】

第1周波数計算部622-iは、情報取得装置602-i内のタイミング信号の周波数sensor\_freqを変換部624-iへ出力する。レジスタ911は、sensor\_freqを記憶し、DSP914へ出力する。

#### 【0064】

10

20

30

40

50

DSP913は、レジスタ906が出力する`1st_pkt_clk`と、レジスタ910が出力する`FPGA_freq`とを用いて、受信部621が最初のパケットを受信したときのOS時刻と、レジスタ905が記憶する`OS_time`との差分を計算する。DSP913は、`1st_pkt_clk`を`FPGA_freq`で除算することで、差分を求めることができる。そして、DSP913は、計算した差分を加算器915へ出力する。

#### 【0065】

加算器915は、DSP913が出力する差分を、レジスタ905が出力する`OS_time`に加算することで、受信部621が最初のパケットを受信したときのOS時刻を求め、加算器916へ出力する。このように、DSP913及び加算器915によって、`1st_pkt_clk`を、最初のパケットの受信時刻を示すOS時刻に変換することができる。

10

#### 【0066】

DSP914は、レジスタ911が出力する`sensor_freq`を用いて、減算器912が出力するカウント値の差分から、受信したパケットの`sensor_cnt`と最初のパケットの`sensor_cnt`との差分に対応する、OS時刻の差分を計算する。DSP914は、カウント値の差分を`sensor_freq`で除算することで、OS時刻の差分求めることができる。そして、DSP914は、計算した差分を加算器916へ出力する。

#### 【0067】

DSP914が出力するOS時刻の差分は、`sensor_cnt`の差分を`sensor_freq`に基づいて変換した値であるため、情報取得装置602-iと時刻同期部613との間におけるパケットの転送時間を含んでいない。したがって、通信ネットワーク603の転送遅延とは無関係である。

20

#### 【0068】

加算器916は、DSP914が出力する差分を、加算器915が出力するOS時刻に加算することで、最初のパケットの受信時刻を示すOS時刻から、`sensor_cnt`の差分に対応する相対時間が経過した後のOS時刻を求める。そして、加算器916は、求めたOS時刻を`Real_time`として付加部625へ出力する。

#### 【0069】

このように、減算器912、DSP914、及び加算器916によって、受信したパケットの`sensor_cnt`を、そのパケットに含まれる情報の取得時刻を示すOS時刻に変換することができる。加算器916が出力する`Real_time`は、受信したパケットに含まれる情報にタイムスタンプとして付加される。

30

#### 【0070】

図10A及び図10Bは、図9の変換部624-iが行う時刻同期処理の具体例を示すフローチャートである。まず、書き込み回路901は、CPU611から`OS_system_time`を受信したか否かをチェックする(ステップ1001)。

#### 【0071】

`OS_system_time`を受信した場合(ステップ1001, YES)、書き込み回路901は、`OS_system_time`を`OS_time`としてレジスタ905に書き込む(ステップ1003)。そして、書き込み回路901は、カウンタ904の`FPGA_count`をクリアし(ステップ1004)、レジスタ909の`first_pkt_flag`をクリアする(ステップ1005)。

40

#### 【0072】

次に、判定回路902は、受信部621から`sensor_valid`を受信したか否かをチェックする(ステップ1002)。`sensor_valid`を受信した場合(ステップ1002, YES)、判定回路902は、レジスタ909の`first_pkt_flag`をチェックする(ステップ1006)。

#### 【0073】

`first_pkt_flag`が論理“0”である場合(ステップ1006, YES)、判

50

定回路902は、`first_pkt_flag`に論理“1”を設定する（ステップ1009）。そして、判定回路902は、カウンタ904が出力する`FPGA_cnt`を、`1st_pkt_clock`としてレジスタ906に書き込む（ステップ1010）。

#### 【0074】

次に、セレクタ903は、`sensor_cnt`をレジスタ907へ出力し、レジスタ907は、`sensor_cnt`を`1st_pkt_clock`として記憶する（ステップ1011）。そして、セレクタ903は、`sensor_cnt`をレジスタ908へ出力し、レジスタ908は、`sensor_cnt`を`curr_pkt_clock`として記憶する（ステップ1007）。

#### 【0075】

次に、減算器912は、`curr_pkt_clock`から`1st_pkt_clock`を減算することで、カウント値の差分を求める（ステップ1008）。そして、DSP914は、レジスタ911の`sensor_freq`を用いて、減算器912が出力するカウント値の差分をOS時刻の差分に変換する（ステップ1012）。

#### 【0076】

次に、DSP913は、レジスタ910の`FPGA_freq`を用いて、レジスタ906の`1st_pkt_clock`を、受信部621が最初のパケットを受信したときのOS時刻と、レジスタ905の`OS_time`との差分に変換する（ステップ1013）。

#### 【0077】

次に、加算器915は、DSP913が出力する差分を、レジスタ905の`OS_time`に加算し、加算器916は、DSP914が出力する差分を、加算器915が出力する加算結果に加算する（ステップ1014）。これにより、`Real_time`が求められる。

#### 【0078】

次に、カウンタ904は、`FPGA_cnt`をカウントアップする（ステップ1015）。そして、変換部624-iは、レジスタ911の`sensor_freq`を更新し（ステップ1016）、レジスタ910の`FPGA_freq`を更新して（ステップ1017）、ステップ1001以降の処理を繰り返す。

#### 【0079】

OS\_system\_timeを受信していない場合（ステップ1001, NO）、変換部624-iは、ステップ1002以降の処理を行う。`first_pkt_flag`が論理“1”である場合（ステップ1006, NO）、変換部624-iは、ステップ1007以降の処理を行う。`sensor_valid`を受信していない場合（ステップ1002, NO）、変換部624-iは、ステップ1015以降の処理を行う。

#### 【0080】

図11は、図9の変換部624-iが行う時刻同期処理の例を示している。CPU611は、所定のタイミングで`OS_system_time`を出力し、レジスタ905の`OS_time`を更新する。

#### 【0081】

図11の例では、15:00:05、15:00:10、及び15:00:16が順番に`OS_time`に設定されている。タイムフレーム1101は、15:00:05から15:00:10までの区間を表し、タイムフレーム1102は、15:00:10から15:00:16までの区間を表す。

#### 【0082】

タイムフレーム1101において、受信部621は、`FPGA_cnt = 50`のときに最初のパケットを受信し、`FPGA_cnt = 71`のときに2番目のパケットを受信し、`FPGA_cnt = 125`のときに3番目のパケットを受信する。最初のパケットの`sensor_cnt`は1010であり、2番目のパケットの`sensor_cnt`は1050であり、3番目のパケットの`sensor_cnt`は1150である。

#### 【0083】

10

20

30

40

50

この場合、2番目のパケットのsensor\_cntと最初のパケットのsensor\_cntとの差分“40”が、OS時刻の差分1111に変換される。そして、FPGA\_cnt = 50に対応するOS時刻に差分1111が加算されて、sensor\_cnt = 1050に対応するReal\_timeが求められる。

#### 【0084】

また、3番目のパケットのsensor\_cntと最初のパケットのsensor\_cntとの差分“140”が、OS時刻の差分1112に変換される。そして、FPGA\_cnt = 50に対応するOS時刻に差分1112が加算されて、sensor\_cnt = 150に対応するReal\_timeが求められる。

#### 【0085】

タイムフレーム1102において、受信部621は、FPGA\_cnt = 36のときに最初のパケットを受信し、FPGA\_cnt = 73のときに2番目のパケットを受信し、FPGA\_cnt = 114のときに3番目のパケットを受信する。最初のパケットのsensor\_cntは1213であり、2番目のパケットのsensor\_cntは1255であり、3番目のパケットのsensor\_cntは1291である。

#### 【0086】

タイムフレーム1102においても、タイムフレーム1101の場合と同様にして、各sensor\_cntに対応するReal\_timeが求められる。

#### 【0087】

図6の情報処理システムによれば、各タイムフレーム内において受信した複数のパケットに含まれる情報に、情報取得装置602-iが情報を取得したときのOS時刻を示すタイムスタンプが付加される。複数の情報それぞれに付加されたタイムスタンプの差分は正確な相対時間を表しているため、それらの情報の取得時刻を情報処理装置601のOS時刻と同期させることができる。この場合、図3の構成のように、情報処理装置601から情報取得装置602-iへリセット信号を送信する必要はない。

#### 【0088】

また、各情報取得装置602-iが個別のタイミング信号に同期して動作している場合であっても、情報取得装置602-1～情報取得装置602-Nの間で、情報の取得時刻を同期させることができることになる。

#### 【0089】

図12は、図6の第2周波数計算部623の構成例を示している。図12の第2周波数計算部623は、書き込み回路1201、ロード回路( )1202、カウンタ1203、レジスタ1204～レジスタ1206、減算器1207、減算器1208、DSP1209、及び統計値計算回路1210を含む。

#### 【0090】

書き込み回路1201は、CPU611からOS\_system\_timeを受信すると、ロード指示をロード回路1202へ出力し、OS\_system\_timeをcurrent\_timeとしてレジスタ1205に書き込む。レジスタ1205は、current\_timeを減算器1208へ出力する。

#### 【0091】

そして、書き込み回路1201は、カウンタ1203をクリアして、カウント値に0を設定する。その後、カウンタ1203は、FPGA\_ckに同期してカウント動作を行い、カウント値をレジスタ1204及び減算器1207へ出力する。

#### 【0092】

ロード回路1202は、書き込み回路1201が出力するロード指示に従って、OS\_system\_timeが書き込まれる直前のcurrent\_timeを、last\_timeとしてレジスタ1206に書き込む。レジスタ1206は、last\_timeを減算器1208へ出力する。

#### 【0093】

また、ロード回路1202は、書き込み回路1201が出力するロード指示に従って、

10

20

30

40

50

カウンタ 1203 がクリアされる直前のカウント値を、レジスタ 1204 に書き込む。レジスタ 1204 は、カウント値を減算器 1207 へ出力する。

#### 【0094】

減算器 1207 は、レジスタ 1204 が出力するカウント値から、カウンタ 1203 が出力するカウント値を減算することで、カウント値の差分を求め、DSP1209 へ出力する。減算器 1208 は、レジスタ 1205 が出力する current\_time から、レジスタ 1206 が出力する last\_time を減算することで、OS 時刻の差分を求め、DSP1209 へ出力する。

#### 【0095】

DSP1209 は、減算器 1207 が出力する差分を、減算器 1208 が出力する差分で除算することで、FPGA\_clk の周波数を求め、統計値計算回路 1210 へ出力する。統計値計算回路 1210 は、DSP1209 が出力する複数の周波数の統計値を求め、求めた統計値を FPGA\_freq として出力する。統計値としては、平均値、中央値、最頻値等を用いることができる。統計値を求める必要がない場合は、統計値計算回路 1210 を省略してもよい。

10

#### 【0096】

図 12 の第 2 周波数計算部 623 によれば、環境の変化によって、時刻同期部 613 内のクロック信号 FPGA\_clk の周波数が変動する場合であっても、正確な周波数を求めることができる。そして、求められた周波数を用いることで、図 9 の Real\_time を補正することができる。

20

#### 【0097】

図 13 は、図 6 の第 1 周波数計算部 622-i の構成例を示している。図 13 の第 1 周波数計算部 622-i は、カウンタ 1301、レジスタ 1302～レジスタ 1304、減算器 1305、減算器 1306、DSP1307、DSP1308、及び統計値計算回路 1309 を含む。

30

#### 【0098】

カウンタ 1301 は、FPGA\_clk に同期してカウント動作を行い、カウント値 FPGA\_cur\_clk をレジスタ 1302 及び減算器 1305 へ出力する。第 1 周波数計算部 622-i は、受信部 621 から sensor\_valid を受信したとき、カウンタ 1301 が出力する FPGA\_cur\_clk を、FPGA\_last\_clk としてレジスタ 1302 に書き込む。レジスタ 1302 は、FPGA\_last\_clk を減算器 1305 へ出力する。

#### 【0099】

レジスタ 1303 は、受信部 621 が出力する sensor\_cnt を、sensor\_cur\_clk として記憶し、sensor\_cur\_clk を減算器 1306 へ出力する。第 1 周波数計算部 622-i は、受信部 621 から sensor\_valid を受信したとき、レジスタ 1303 の sensor\_cur\_clk を、sensor\_last\_clk としてレジスタ 1304 に書き込む。レジスタ 1304 は、sensor\_last\_clk を減算器 1306 へ出力する。

40

#### 【0100】

減算器 1305 は、カウンタ 1301 が出力する FPGA\_cur\_clk から、レジスタ 1302 が出力する FPGA\_last\_clk を減算することで、カウント値の差分を求め、DSP1307 へ出力する。減算器 1306 は、レジスタ 1303 が出力する sensor\_cur\_clk から、レジスタ 1304 が出力する sensor\_last\_clk を減算することで、sensor\_cnt の差分を求め、DSP1307 へ出力する。

#### 【0101】

DSP1307 は、減算器 1306 が出力する差分を、減算器 1305 が出力する差分で除算し、除算結果を DSP1308 へ出力する。DSP1308 は、DSP1307 が出力する除算結果に、第 2 周波数計算部 623 が出力する FPGA\_freq を乗算する

50

ことで、情報取得装置 602-i 内のタイミング信号の周波数を求め、統計値計算回路 1309 へ出力する。統計値計算回路 1309 は、DSP1308 が output する複数の周波数の統計値を求め、求めた統計値を sensor\_freq として出力する。統計値としては、平均値、中央値、最頻値等を用いることができる。統計値を求める必要がない場合は、統計値計算回路 1309 を省略してもよい。

#### 【0102】

図 13 の第 1 周波数計算部 622-i によれば、環境の変化によって、情報取得装置 602-i 内のタイミング信号の周波数が変動する場合であっても、正確な周波数を求めることが可能である。そして、求められた周波数を用いることで、図 9 の Real\_time を補正することが可能になる。

10

#### 【0103】

図 14 は、図 12 の第 2 周波数計算部 623 によって計算される、クロック信号 FPG\_A\_clk の周波数 FPG\_A\_freq の例を示している。この例では、統計値計算回路 1210 による統計値計算は省略されている。図 14 の Sub 出力は、減算器 1207 が出力するカウント値の差分を表す。FPG\_A\_freq は、次式により計算される。

$$\text{FPG_A\_freq} = \text{Sub 出力} / (\text{current\_time} - \text{last\_time}) \quad (1)$$

#### 【0104】

図 11 の時刻同期処理において、OS\_time = 15:00:05 のとき、Sub 出力 = 11000 である。次に、OS\_time = 15:00:10 のとき、カウンタ 1203、レジスタ 1204、current\_time、及び last\_time の値は、以下のようになる。

20

```
カウンタ1203 00000
レジスタ1204 10000
current_time 15:00:10
last_time 15:00:05
```

#### 【0105】

この場合、Sub 出力 = 10000 となり、FPG\_A\_freq は、次式により計算される。

$$\text{FPG_A\_freq} = 10000 / (10 - 5) = 2000 [\text{Hz}] \quad (2)$$

30

#### 【0106】

次に、OS\_time = 15:00:16 のとき、カウンタ 1203、レジスタ 1204、current\_time、及び last\_time の値は、以下のようになる。

```
カウンタ1203 00000
レジスタ1204 11000
current_time 15:00:16
last_time 15:00:10
```

#### 【0107】

この場合、Sub 出力 = 11000 となり、FPG\_A\_freq は、次式により計算される。

$$\text{FPG_A\_freq} = 11000 / (16 - 10) = 1833 [\text{Hz}] \quad (3)$$

40

#### 【0108】

図 15 は、図 13 の第 1 周波数計算部 622-i によって計算される、タイミング信号の周波数 sensor\_freq の例を示している。この例では、統計値計算回路 1309 による統計値計算は省略されている。sensor\_freq は、次式により計算される。

$$\begin{aligned} \text{sensor\_freq} &= \{ (\text{sensor\_cur\_clk} - \text{sensor\_last\_clk}) \\ &/ (\text{FPG_A\_cur\_clk} - \text{FPG_A\_last\_clk}) \} \\ &\times \text{FPG_A\_freq} \end{aligned} \quad (4)$$

50

## 【0109】

図11のタイムフレーム1101において、最初のパケットを受信したとき、`sensor_cnt`、`FPGA_cur_clk`、及び`FPGA_freq`の値は、以下のようになる。

```
sensor_cnt = 1010
FPGA_cur_clk = 2050
FPGA_freq = 2000
```

## 【0110】

次に、2番目のパケットを受信したとき、`sensor_cnt`、`FPGA_cur_clk`、`FPGA_last_clk`、`sensor_cur_clk`、`sensor_last_clk`、及び`FPGA_freq`の値は、以下のようになる。 10

```
sensor_cnt = 1050
FPGA_cur_clk = 2071
FPGA_last_clk = 2050
sensor_cur_clk = 1050
sensor_last_clk = 1010
FPGA_freq = 2000
```

## 【0111】

この場合、`sensor_freq`は、次式により計算される。

```
sensor_freq
= { (1050 - 1010) / (2071 - 2050) } * 2000
= 3810 [Hz] (5)
```

## 【0112】

次に、3番目のパケットを受信したとき、`sensor_cnt`、`FPGA_cur_clk`、`FPGA_last_clk`、`sensor_cur_clk`、`sensor_last_clk`、及び`FPGA_freq`の値は、以下のようになる。 30

```
sensor_cnt = 1150
FPGA_cur_clk = 2125
FPGA_last_clk = 2071
sensor_cur_clk = 1150
sensor_last_clk = 1050
FPGA_freq = 2000
```

## 【0113】

この場合、`sensor_freq`は、次式により計算される。

```
sensor_freq
= { (1150 - 1050) / (2125 - 2071) } * 2000
= 3704 [Hz] (6)
```

## 【0114】

図16は、図9の変換部624-iによって計算されるOS時刻`Real_time`の例を示している。`Real_time`は、次式により計算される。 40

```
Real_time
= (cur_pkt_cnt - 1st_pkt_cnt) / sensor_freq
+ 1st_pkt_clk / FPGA_freq + OS_time
(7)
```

## 【0115】

図11の時刻同期処理において、`OS_time = 15:00:05`のとき、`1st_pkt_clk`、`first_pkt_flag`、`FPGA_freq`、及び`sensor_freq`の値は、以下のようになる。

```
1st_pkt_clk = 0
first_pkt_flag = 0
```

10

20

30

40

50

```
FPGA_freq = 2000
sensor_freq = 4000
【0116】
```

次に、最初のパケットを受信したとき、sensor\_cnt、1st\_pkt\_clk、first\_pkt\_flag、1st\_pkt\_cnt、cur\_pkt\_cnt、FPGA\_freq、及びsensor\_freqの値は、以下のようになる。

```
sensor_cnt = 1010
1st_pkt_clk = 50
first_pkt_flag = 1
1st_pkt_cnt = 1010
cur_pkt_cnt = 1010
FPGA_freq = 2000
sensor_freq = 4000
【0117】
```

この場合、Real\_timeは、次式により計算される。

```
Real_time
= (1010 - 1010) / 4000 + 50 / 2000 + 15 : 00 : 05
= 15 : 00 : 05 : 025
(8)
```

【0118】

次に、2番目のパケットを受信したとき、sensor\_cnt、1st\_pkt\_clk、first\_pkt\_flag、1st\_pkt\_cnt、cur\_pkt\_cnt、FPGA\_freq、及びsensor\_freqの値は、以下のようになる。

```
sensor_cnt = 1050
1st_pkt_clk = 50
first_pkt_flag = 1
1st_pkt_cnt = 1010
cur_pkt_cnt = 1050
FPGA_freq = 2000
sensor_freq = 3810
【0119】
```

この場合、Real\_timeは、次式により計算される。

```
Real_time
= (1050 - 1010) / 3810 + 50 / 2000 + 15 : 00 : 05
= 15 : 00 : 05 : 035
(9)
```

【0120】

次に、3番目のパケットを受信したとき、sensor\_cnt、1st\_pkt\_clk、first\_pkt\_flag、1st\_pkt\_cnt、cur\_pkt\_cnt、FPGA\_freq、及びsensor\_freqの値は、以下のようになる。

```
sensor_cnt = 1150
1st_pkt_clk = 50
first_pkt_flag = 1
1st_pkt_cnt = 1010
cur_pkt_cnt = 1150
FPGA_freq = 2000
sensor_freq = 3704
【0121】
```

この場合、Real\_timeは、次式により計算される。

```
Real_time
= (1150 - 1010) / 3704 + 50 / 2000 + 15 : 00 : 05
= 15 : 00 : 05 : 063
(10)
```

10

20

30

40

50

**【0122】**

ところで、通信ネットワーク603において障害等が発生した場合、情報取得装置602-iから送信されたセンサデータのパケットが消失して、情報処理装置601に到達しないことがある。このような場合であっても、時刻同期部613は、情報処理装置601に到達したパケットに含まれる情報に対して、タイムスタンプを付加することが可能である。

**【0123】**

図17は、通信ネットワーク603におけるパケット消失時の時刻同期処理の例を示している。図17(a)は、図11のタイムフレーム1101において、最初のパケットが消失した場合の時刻同期処理の例を示している。  
10

**【0124】**

図17(a)のタイムフレーム1101において、最初のパケットが消失した場合、受信部621は、FPGA\_cnt = 71のときに2番目のパケットを受信し、FPGA\_cnt = 125のときに3番目のパケットを受信する。

**【0125】**

この場合、3番目のパケットのsensor\_cntと2番目のパケットのsensor\_cntとの差分“100”が、OS時刻の差分1701に変換される。そして、FPGA\_cnt = 71に対応するOS時刻に差分1701が加算されて、sensor\_cnt = 1150に対応するReal\_timeが求められる。

**【0126】**

図17(b)は、図11のタイムフレーム1101において、2番目のパケットが消失した場合の時刻同期処理の例を示している。  
20

**【0127】**

図17(b)のタイムフレーム1101において、2番目のパケットが消失した場合、受信部621は、FPGA\_cnt = 50のときに最初のパケットを受信し、FPGA\_cnt = 125のときに3番目のパケットを受信する。

**【0128】**

この場合、3番目のパケットのsensor\_cntと最初のパケットのsensor\_cntとの差分“140”が、OS時刻の差分1112に変換される。そして、FPGA\_cnt = 50に対応するOS時刻に差分1112が加算されて、sensor\_cnt = 1150に対応するReal\_timeが求められる。  
30

**【0129】**

このように、図6の情報処理システムによれば、タイムフレーム内のいずれのパケットが消失した場合であっても、いずれかのパケットを受信したOS時刻を基準として、後続するパケットに含まれる情報にタイムスタンプを付加することができる。

**【0130】**

図18は、ソフトウェア制御による時刻同期部613の構成例を示している。図18の時刻同期部613は、CPU1801、メモリ1802、及びインターフェース1803を含むコンピュータである。

**【0131】**

メモリ1802は、例えば、ROM(Read Only Memory)、RAM(Random Access Memory)、フラッシュメモリ等の半導体メモリであり、処理に用いられるプログラム及びデータを格納する。メモリ1802は、図4の記憶部411として用いることができる。  
40

**【0132】**

CPU1801(プロセッサ)は、例えば、メモリ1802を利用してプログラムを実行することにより、図4の受信時刻変換部413、時刻情報変換部414、及び付加部415として動作する。CPU1801は、メモリ1802を利用してプログラムを実行することにより、第1周波数計算部、第2周波数計算部、減算部、第1差分計算部、第1加算部、第2差分計算部、及び第2加算部としても動作する。  
50

**【 0 1 3 3 】**

インターフェース 1803 は、通信ネットワーク 603 に接続され、通信に伴うデータ変換を行う通信インターフェース回路である。時刻同期部 613 は、プログラム及びデータを外部の装置からインターフェース 1803 を介して受信し、それらをメモリ 1802 にロードして使用することができる。インターフェース 1803 は、図 4 の受信部 412 又は図 6 の受信部 621 として用いることができる。

**【 0 1 3 4 】**

図 18 の時刻同期部 613 は、さらに、可搬型記録媒体を駆動する媒体駆動装置を含んでいてもよい。可搬型記録媒体は、メモリデバイス、フレキシブルディスク、光ディスク、光磁気ディスク等である。可搬型記録媒体は、CD-ROM (Compact Disk Read Only Memory)、DVD (Digital Versatile Disk)、USB (Universal Serial Bus) メモリ等であってもよい。オペレータ又はユーザは、この可搬型記録媒体にプログラム及びデータを格納しておき、それらをメモリ 1802 にロードして使用することができる。10

**【 0 1 3 5 】**

このように、処理に用いられるプログラム及びデータを格納するコンピュータ読み取り可能な記録媒体は、メモリ 1802 又は可搬型記録媒体のような、物理的な（非一時的な）記録媒体である。

**【 0 1 3 6 】**

図 4 の情報処理装置 401 の構成は一例に過ぎず、情報処理装置 401 の用途又は条件に応じて一部の構成要素を省略又は変更してもよい。20

**【 0 1 3 7 】**

図 6 の情報処理システムの構成は一例に過ぎず、情報処理システムの用途又は条件に応じて一部の構成要素を省略又は変更してもよい。例えば、時刻同期部 613 内のクロック信号の周波数があまり変動せず、第 1 周波数計算部 622-i 及び変換部 624-i がその周波数を示す固定値を保持している場合は、第 2 周波数計算部 623 を省略することができる。情報取得装置 602-i 内のタイミング信号の周波数があまり変動せず、変換部 624-i がその周波数を示す固定値を保持している場合は、第 1 周波数計算部 622-i を省略することができる。

**【 0 1 3 8 】**

情報取得装置 602-1 ~ 情報取得装置 602-N は、情報処理装置 601 内に設けられてもよい。30

**【 0 1 3 9 】**

図 7 のセンサモジュール 701 及び図 8 のビデオモジュール 801 の構成は一例に過ぎず、情報処理システムの用途又は条件に応じて一部の構成要素を省略又は変更してもよい。図 7 の圧力センサ 711 の代わりに、温度センサ、湿度センサ、風力センサ等を用いることもできる。

**【 0 1 4 0 】**

図 9 の変換部 624-i、図 12 の第 2 周波数計算部 623、図 13 の第 1 周波数計算部 622-i、及び図 18 の時刻同期部 613 の構成は一例に過ぎず、情報処理システムの用途又は条件に応じて一部の構成要素を省略又は変更してもよい。40

**【 0 1 4 1 】**

図 5、図 10A、及び図 10B のフローチャートは一例に過ぎず、変換部 624-i の構成又は条件に応じて一部の処理を省略又は変更してもよい。

**【 0 1 4 2 】**

図 11 及び図 17 に示した時刻同期処理は一例に過ぎず、sensor\_cnt、FPG\_A\_cnt、及び OS\_time は、情報取得装置 602-i が取得する情報に応じて変化する。図 14 ~ 図 16 に示した計算結果は一例に過ぎず、計算結果は、情報取得装置 602-i が取得する情報に応じて変化する。

**【 0 1 4 3 】**

式(1) ~ 式(10)の計算式は一例に過ぎず、情報処理装置 601 の構成又は条件に50

応じて別の計算式を用いてもよい。

**【0144】**

開示の実施形態とその利点について詳しく説明したが、当業者は、特許請求の範囲に明確に記載した本発明の範囲から逸脱することなく、様々な変更、追加、省略をすることができるであろう。

**【0145】**

図2乃至図18を参照しながら説明した実施形態に関し、さらに以下の付記を開示する。

(付記1)

第1システム時刻を記憶する記憶部と、

前記記憶部に前記第1システム時刻が書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報とを受信し、前記第1情報と前記第1時刻情報とを受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報とを受信する受信部と、

前記第1システム時刻に基づいて、前記受信部が前記第1情報と前記第1時刻情報を受信した受信時刻を、第2システム時刻に変換する受信時刻変換部と、

前記第2システム時刻と前記第1時刻情報をに基づいて、前記第2時刻情報を第3システム時刻に変換する時刻情報変換部と、

前記第2システム時刻を前記第1情報に付加し、前記第3システム時刻を前記第2情報を付加する付加部と、

を備えることを特徴とする情報処理装置。

(付記2)

前記時刻情報変換部は、

前記第2時刻情報と前記第1時刻情報との差分を生成する減算部と、

前記第2時刻情報と前記第1時刻情報との差分から、前記第3システム時刻と前記第2システム時刻との差分を計算する第1差分計算部と、

前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、前記第3システム時刻を求める第1加算部と、  
を含むことを特徴とする付記1記載の情報処理装置。

(付記3)

前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算する第1周波数計算部をさらに備え、

前記受信時刻変換部は、前記受信部が前記第1情報と前記第1時刻情報を受信したときに、前記クロック信号に基づいて特定された時刻を、前記受信時刻として用い、

前記第1差分計算部は、前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算することを特徴とする付記2記載の情報処理装置。

(付記4)

前記受信時刻変換部は、

前記受信時刻と前記クロック信号の周波数とを用いて、前記第2システム時刻と前記第1システム時刻との差分を計算する第2差分計算部と、

前記第2システム時刻と前記第1システム時刻との差分を前記第1システム時刻に加算することで、前記第2システム時刻を求める第2加算部と、  
を含むことを特徴とする付記3記載の情報処理装置。

(付記5)

前記第1システム時刻と前記クロック信号とを用いて、前記クロック信号の周波数を計算する第2周波数計算部をさらに備えることを特徴とする付記4記載の情報処理装置。

(付記6)

前記受信部は、前記記憶部に前記第1システム時刻が書き込まれた後に、複数の情報取得装置のうちいずれかの情報取得装置から、前記いずれかの情報取得装置が取得した第1

10

20

30

40

50

情報と第1時刻情報を受信し、前記第1情報と前記第1時刻情報を受信した後に、前記いずれかの情報取得装置から、第2情報と第2時刻情報を受信し、

前記付加部は、前記第2システム時刻を前記いずれかの情報取得装置から受信した第1情報に付加し、前記第3システム時刻を前記いずれかの情報取得装置から受信した第2情報に付加することを特徴とする付記1乃至5のいずれか1項に記載の情報処理装置。

(付記7)

情報処理装置によって実行される時刻同期方法であって、

前記情報処理装置が、

第1システム時刻が記憶部に書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報を受信し、

10

前記第1情報と前記第1時刻情報を受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報を受信し、

前記第1システム時刻に基づいて、前記第1情報と前記第1時刻情報を受信した受信時刻を、第2システム時刻に変換し、

前記第2システム時刻と前記第1時刻情報を基づいて、前記第2時刻情報を第3システム時刻に変換し、

前記第2システム時刻を前記第1情報を付加し、

前記第3システム時刻を前記第2情報を付加する、

ことを特徴とする時刻同期方法。

(付記8)

20

前記情報処理装置は、

前記第2時刻情報と前記第1時刻情報との差分を生成し、

前記第2時刻情報と前記第1時刻情報との差分から、前記第3システム時刻と前記第2システム時刻との差分を計算し、

前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、前記第3システム時刻を求める特徴とする付記7記載の時刻同期方法。

(付記9)

前記情報処理装置は、

前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算し、

30

前記第1情報と前記第1時刻情報を受信したときに、前記クロック信号に基づいて特定された時刻を、前記受信時刻として用い、

前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算することを特徴とする付記8記載の時刻同期方法。

(付記10)

前記情報処理装置は、

前記受信時刻と前記クロック信号の周波数とを用いて、前記第2システム時刻と前記第1システム時刻との差分を計算し、

40

前記第2システム時刻と前記第1システム時刻との差分を前記第1システム時刻に加算することで、前記第2システム時刻を求める特徴とする付記9記載の時刻同期方法。

(付記11)

第1システム時刻が記憶部に書き込まれた後に、情報取得装置から、前記情報取得装置が取得した第1情報と、前記第1情報の取得時刻を示す第1時刻情報を受信し、

前記第1情報と前記第1時刻情報を受信した後に、前記情報取得装置から、第2情報と前記第2情報の取得時刻を示す第2時刻情報を受信し、

前記第1システム時刻に基づいて、前記第1情報と前記第1時刻情報を受信した受信時刻を、第2システム時刻に変換し、

前記第2システム時刻と前記第1時刻情報を基づいて、前記第2時刻情報を第3シス

50

テム時刻に変換し、  
 前記第2システム時刻を前記第1情報に付加し、  
 前記第3システム時刻を前記第2情報に付加する、  
 処理をコンピュータに実行させるための時刻同期プログラム。

(付記12)

前記コンピュータは、  
 前記第2時刻情報と前記第1時刻情報との差分を生成し、  
 前記第2時刻情報と前記第1時刻情報との差分から、前記第3システム時刻と前記第2システム時刻との差分を計算し、  
 前記第3システム時刻と前記第2システム時刻との差分を前記第2システム時刻に加算することで、前記第3システム時刻を求める特徴とする付記11記載の時刻同期プログラム。

(付記13)

前記コンピュータは、  
 前記第2時刻情報と前記第1時刻情報との差分と、クロック信号と、前記クロック信号の周波数とを用いて、前記情報取得装置において取得時刻を生成するタイミング信号の周波数を計算し、

前記第1情報と前記第1時刻情報を受信したときに、前記クロック信号に基づいて特定された時刻を、前記受信時刻として用い、

前記第2時刻情報と前記第1時刻情報との差分と、前記タイミング信号の周波数とを用いて、前記第3システム時刻と前記第2システム時刻との差分を計算することを特徴とする付記12記載の時刻同期プログラム。

(付記14)

前記コンピュータは、  
 前記受信時刻と前記クロック信号の周波数とを用いて、前記第2システム時刻と前記第1システム時刻との差分を計算し、

前記第2システム時刻と前記第1システム時刻との差分を前記第1システム時刻に加算することで、前記第2システム時刻を求める特徴とする付記13記載の時刻同期プログラム。

#### 【符号の説明】

##### 【0146】

101、401、601 情報処理装置

102-1、102-2 センサ

121-1～121-4、122-1～122-4 データ

111、201、301 FPGA

202 OS

211 割り込みコントローラ

212、311 MMIOレジスタ

213 検出部

221 OS時刻

312 加算部

402、602-1～602-N 情報取得装置

411 記憶部

412、621 受信部

413 受信時刻変換部

414 時刻情報変換部

415、625 付加部

603 通信ネットワーク

611 CPU

612 メモリ

10

20

30

40

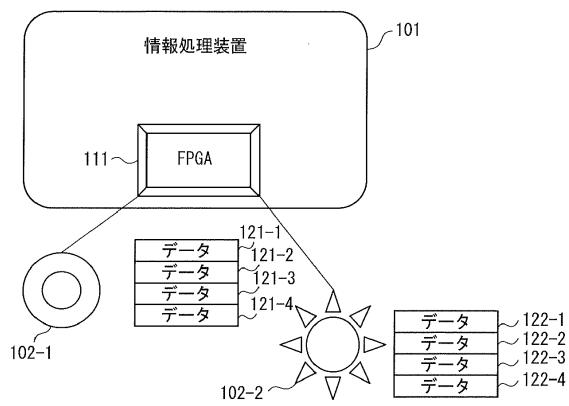
50

6 1 3 時刻同期部  
 6 2 2 - 1 ~ 6 2 2 - N 第 1 周波数計算部  
 6 2 3 第 2 周波数計算部  
 6 2 4 - 1 ~ 6 2 4 - N 変換部  
 7 0 1 センサモジュール  
 7 1 1 圧力センサ  
 7 1 2、8 1 2 水晶振動子  
 7 1 3、8 1 3 A D コンバータ  
 7 1 4、8 1 4、9 0 4、1 2 0 3、1 3 0 1 カウンタ  
 7 1 5、8 1 5、9 0 1、1 2 0 1 書き込み回路  
 7 1 6、8 1 6 データバッファ  
 7 1 7 I 2 C バスマスター  
 7 2 1 情報  
 8 0 1 ビデオモジュール  
 8 1 1 撮像装置  
 8 2 1 画像情報  
 9 0 2 判定回路  
 9 0 3 セレクタ  
 9 0 5 ~ 9 1 1、1 2 0 4 ~ 1 2 0 6、1 3 0 2 ~ 1 3 0 4 レジスタ  
 9 1 2、1 2 0 7、1 2 0 8、1 3 0 5、1 3 0 6 減算器  
 9 1 3、9 1 4、1 2 0 9、1 3 0 7、1 3 0 8 D S P  
 9 1 5、9 1 6 加算器  
 1 1 0 1、1 1 0 2 タイムフレーム  
 1 1 1 1、1 1 1 2、1 7 0 1 O S 時刻の差分  
 1 2 0 2 ロード回路  
 1 2 1 0、1 3 0 9 統計値計算回路  
 1 8 0 1 C P U  
 1 8 0 2 メモリ  
 1 8 0 3 インタフェース

【図面】

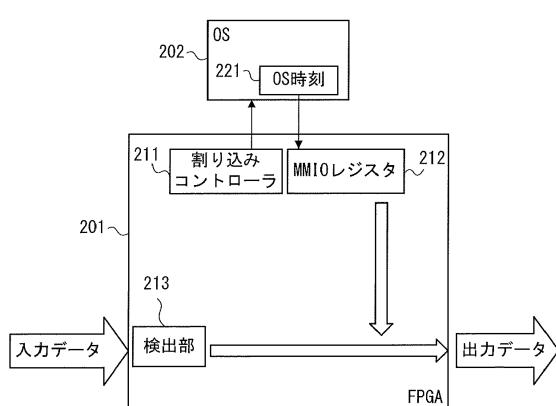
【図 1】

従来の情報処理システムの構成図



【図 2】

FPGAの第1の比較例を示す図



10

20

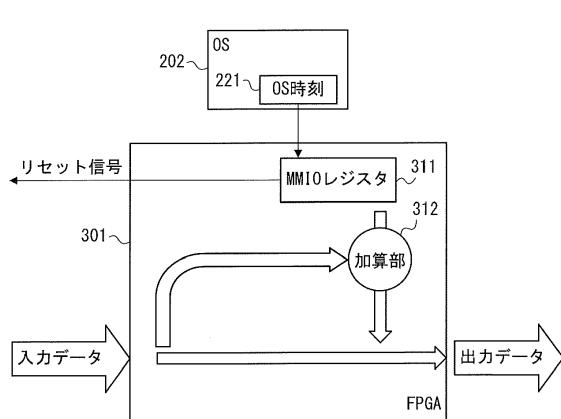
30

40

50

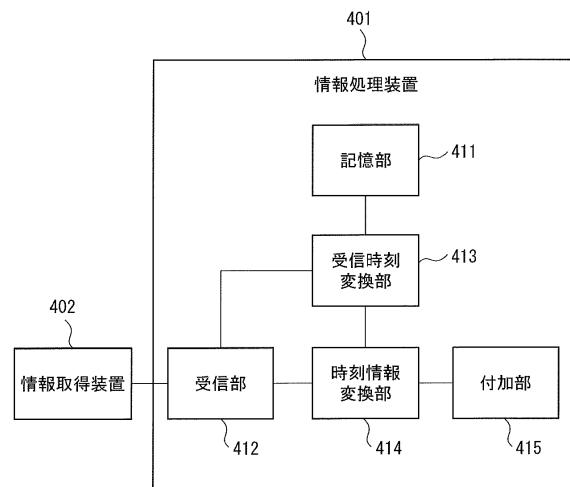
【図3】

FPGAの第2の比較例を示す図



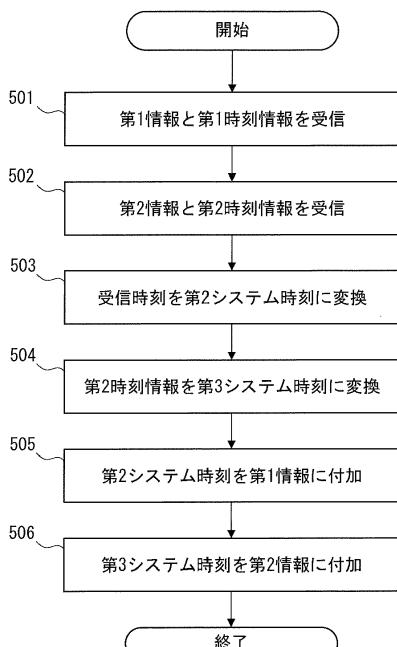
【図4】

情報処理装置の機能的構成図



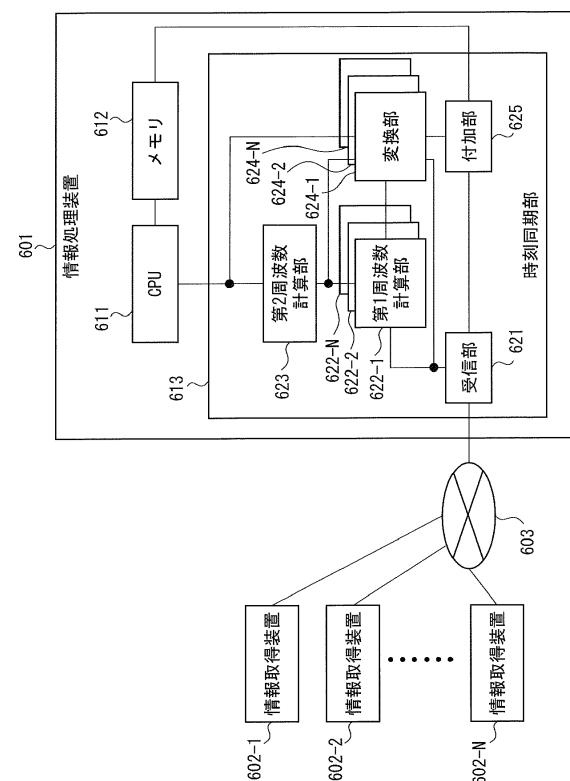
【図5】

時刻同期処理のフローチャート



【図6】

実施形態の情報処理システムの構成図



10

20

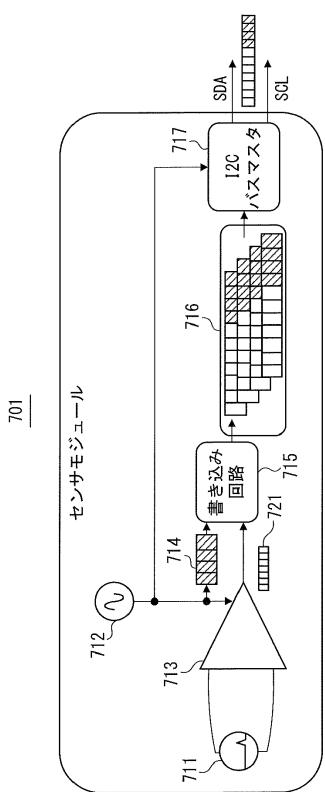
30

40

50

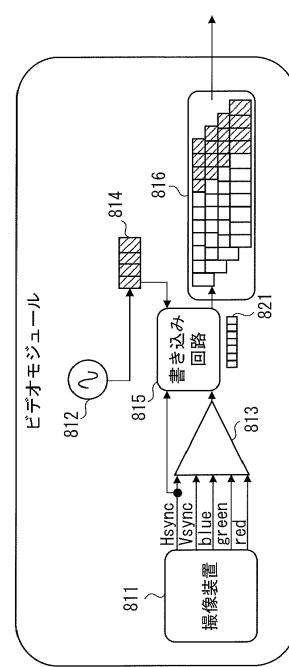
【図 7】

センサモジュールの構成図



【図 8】

ビデオモジュールの構成図

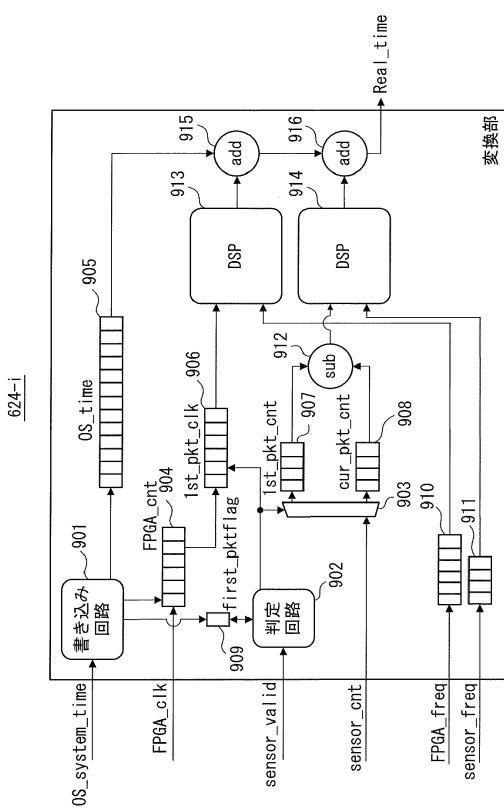


10

20

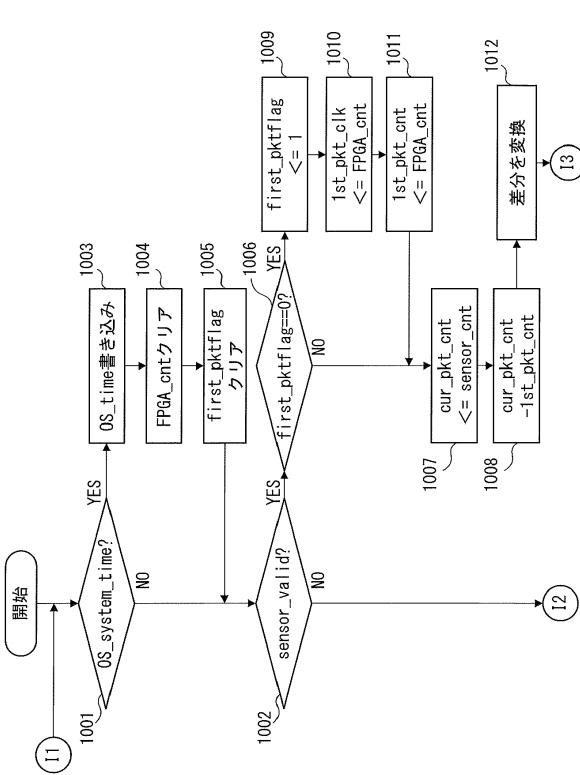
【図 9】

変換部の構成図



【図 10A】

時刻同期処理の具体例を示すフローチャート(その1)



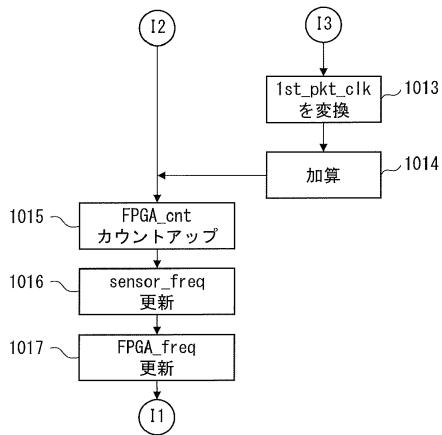
30

40

50

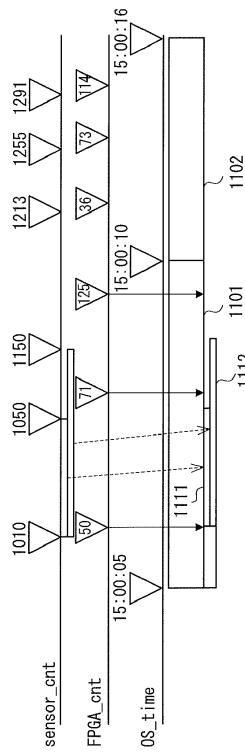
【図 10B】

#### 時刻同期処理の具体例を示すフローチャート(その2)



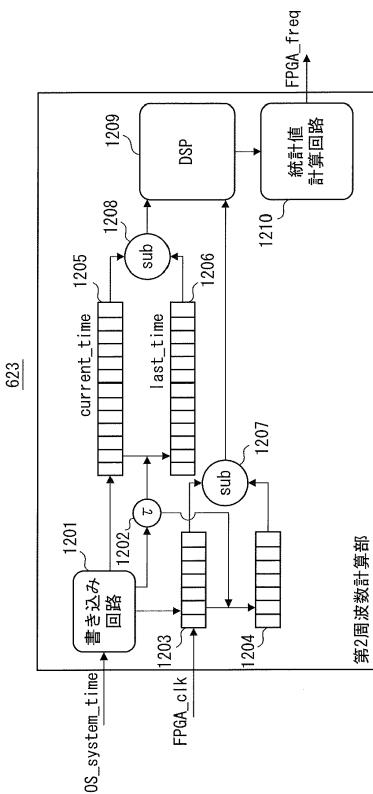
【図 1 1】

## 時刻同期処理を示す図



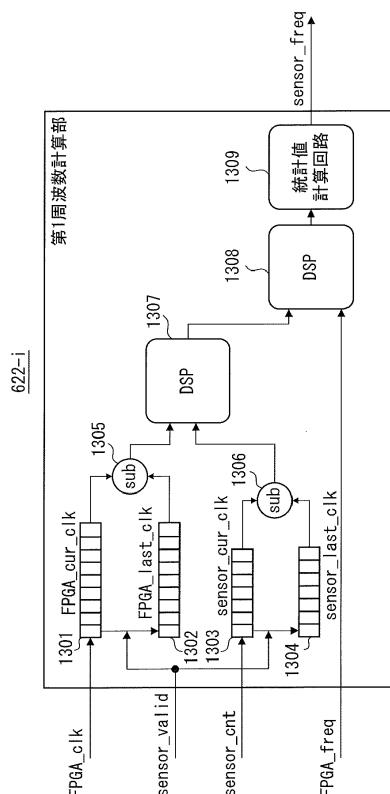
【図12】

## 第2周波数計算部の構成図



【図13】

## 第1周波数計算部の構成図



【図 1 4】

クロック信号の周波数を示す図

OS_time	Sub出力	FPGA_freq
15:00:05	11000	
15:00:10	10000	$10000/(10-5)=2000$
15:00:16	11000	$11000/(16-10)=1833$

【図 1 5】

タイミング信号の周波数を示す図

sensor_cnt	FPGA_cur_clk	FPGA_freq	sensor_freq
1010	2050	2000	$\{((1050-1010)/(2071-2050))*2000=3810$
1050	2071	2000	$\{((1050-1010)/(2071-2050))*2000=3810$
1150	2125	2000	$\{((1150-1050)/(2125-2071))*2000=3704$

10

20

30

40

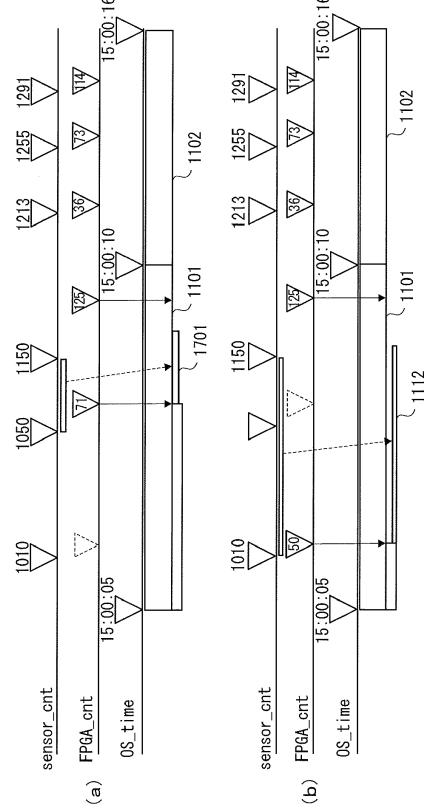
【図 1 6】

OS時刻を示す図

OS_time	sensor_cnt	1st_pkt_clk	first_pkt_flag	FPGA_freq	sensor_freq	Real_time
15:00:05	-	0	0	2000	4000	
1010	50	1	2000	4000	$50/2000+15:00:05=15:00:05:025$	
1050	50	1	2000	3810	$((1050-1010)/3810)*50/2000+15:00:05=15:00:05:035$	
1150	50	1	2000	3704	$((1150-1010)/3704)*50/2000+15:00:05=15:00:05:063$	

【図 1 7】

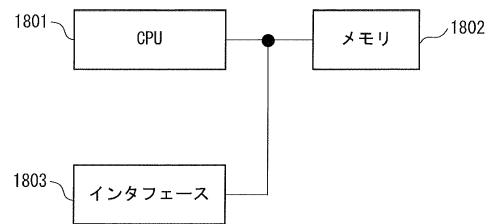
パケット消失時の時刻同期処理を示す図



50

## 【図 1 8】

ソフトウェア制御による時刻同期部の構成図



10

20

30

40

50

---

フロントページの続き

(72)発明者 横山 乾

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 村川 博

大阪府大阪市中央区城見二丁目2番53号 富士通関西中部ネットテック株式会社内

審査官 榎永 雅夫

(56)参考文献 特開2008-209995 (JP, A)

(58)調査した分野 (Int.Cl., DB名)

G 04 G 5 / 00

G 04 G 7 / 00

H 04 Q 9 / 00 - 9 / 16

H 04 M 11 / 00 - 11 / 10