

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-244316

(P2010-244316A)

(43) 公開日 平成22年10月28日(2010.10.28)

(51) Int.Cl.			F I			テーマコード (参考)	
G06F	9/48	(2006.01)	G06F	9/46	452Z	5C159	
H04N	1/41	(2006.01)	H04N	1/41	Z	5C178	
H04N	7/26	(2006.01)	H04N	7/13	Z	5J064	
H03M	7/30	(2006.01)	H03M	7/30	Z		

審査請求 未請求 請求項の数 17 O L (全 45 頁)

(21) 出願番号 特願2009-92683 (P2009-92683)
 (22) 出願日 平成21年4月7日(2009.4.7)

(71) 出願人 000002185
 ソニー株式会社
 東京都港区港南1丁目7番1号
 (74) 代理人 100082131
 弁理士 稲本 義雄
 (74) 代理人 100121131
 弁理士 西川 孝
 (72) 発明者 福原 隆浩
 東京都港区港南1丁目7番1号 ソニー株式会社社内
 (72) 発明者 安藤 勝俊
 東京都港区港南1丁目7番1号 ソニー株式会社社内

最終頁に続く

(54) 【発明の名称】 符号化装置および方法、並びに、復号装置および方法

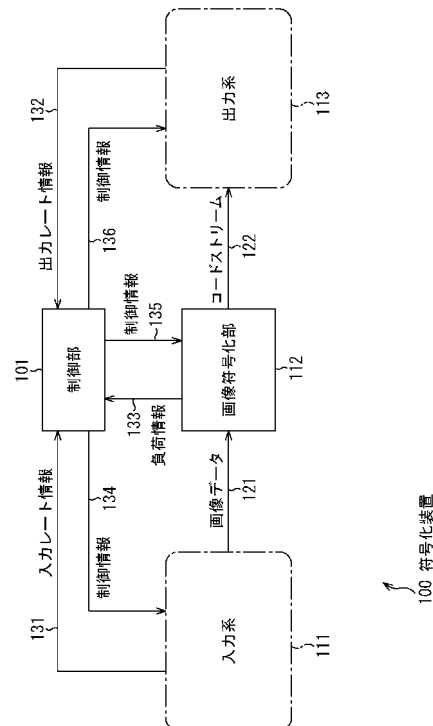
(57) 【要約】

【課題】符号化処理または復号処理においてハードウェア資源をより有効に活用することができるようにする。

【解決手段】制御部101は、画像符号化部112の画像符号化処理の処理速度が入力系111の処理速度より速い場合、画像符号化処理の処理速度を減速させ、入力系111の処理速度に合わせる。また、制御部101は、画像符号化部112の画像符号化処理の処理速度が出力系113の処理速度より速い場合、画像符号化処理の処理速度を減速させ、画像符号化部112の画像符号化処理の処理速度が出力系113の処理速度より遅い場合、画像符号化処理の処理速度を加速させることにより、画像符号化処理の処理速度を出力系113の処理速度に合わせる。本発明は、例えば、符号化装置および復号装置に適用することができる。

【選択図】 図1

図1



【特許請求の範囲】**【請求項 1】**

符号化処理の対象である画像データに対して、前記符号化処理の前に所定の第 1 の処理を行う第 1 の処理手段と、

前記第 1 の処理手段により前記第 1 の処理が行われた前記画像データに対して前記符号化処理を行い、コードストリームを生成する符号化手段と、

前記符号化手段による前記符号化処理により前記画像データが符号化されて生成された前記コードストリームに対して、所定の第 2 の処理を行う第 2 の処理手段と、

前記第 1 の処理手段による前記第 1 の処理の処理速度、および、前記第 2 の処理手段による前記第 2 の処理の処理速度に応じて、前記符号化手段による前記符号化処理の処理速度を制御する制御手段と

を備える符号化装置。

10

【請求項 2】

前記制御手段は、前記符号化処理の処理速度が、前記第 1 の処理の処理速度よりも速い場合、前記符号化処理の処理速度を低下させる

請求項 1 に記載の符号化装置。

【請求項 3】

前記制御手段は、前記符号化処理の処理速度が、前記第 2 の処理の処理速度よりも速い場合、前記符号化処理の処理速度を低下させる

請求項 1 に記載の符号化装置。

20

【請求項 4】

前記制御手段は、前記符号化処理の処理速度が、前記第 2 の処理の処理速度よりも遅い場合、前記符号化処理の処理速度を上昇させる

請求項 1 に記載の符号化装置。

【請求項 5】

前記符号化手段は、複数のコアを持つプロセッサによって実現され、

前記制御手段は、前記符号化処理に割り当てる前記コアの数を増減させることにより、前記符号化処理の処理速度を制御する

請求項 1 に記載の符号化装置。

【請求項 6】

前記符号化手段は、可逆符号化方式により前記符号化処理を行う

請求項 1 に記載の符号化装置。

30

【請求項 7】

前記符号化手段は、JPEG2000方式の可逆符号化方式により前記符号化処理を行い、

前記制御手段は、前記画像データのコードブロック毎のゼロビットプレーン数を取得し、前記ゼロビットプレーン数から有効ビットプレーン数のピクチャ毎の総和を算出し、前記有効ビットプレーン数のピクチャ毎の総和を用いて、前記符号化手段による前記符号化処理の処理速度を制御する

請求項 1 に記載の符号化装置。

【請求項 8】

前記制御手段による前記符号化処理の処理速度の制御に関する負荷制御情報を、前記コードストリームに付加する付加手段をさらに備える

請求項 1 に記載の符号化装置。

40

【請求項 9】

符号化装置の第 1 の処理手段が、符号化処理の対象である画像データに対して、前記符号化処理の前に所定の第 1 の処理を行い、

前記符号化装置の符号化手段が、前記第 1 の処理が行われた前記画像データに対して前記符号化処理を行い、コードストリームを生成し、

前記符号化装置の第 2 の処理手段が、前記符号化処理により前記画像データが符号化されて生成された前記コードストリームに対して、所定の第 2 の処理を行い、

50

前記符号化装置の制御手段が、前記第 1 の処理の処理速度および前記第 2 の処理の処理速度に応じて、前記符号化処理の処理速度を制御する
符号化方法。

【請求項 10】

復号処理の対象である、画像データが符号化されて生成されたコードストリームに対して、前記復号処理の前に所定の第 1 の処理を行う第 1 の処理手段と、

前記第 1 の処理手段により前記第 1 の処理が行われた前記コードストリームに対して前記復号処理を行い、前記画像データを生成する復号手段と、

前記復号手段による前記復号処理により前記コードストリームが復号されて生成された前記画像データに対して、所定の第 2 の処理を行う第 2 の処理手段と、

前記第 1 の処理手段による前記第 1 の処理の処理速度、および、前記第 2 の処理手段による前記第 2 の処理の処理速度に応じて、前記復号手段による前記復号処理の処理速度を制御する制御手段と

を備える復号装置。

【請求項 11】

前記制御手段は、前記復号処理の処理速度が、前記第 1 の処理の処理速度よりも遅い場合、前記復号処理の処理速度を上昇させる

請求項 10 に記載の復号装置。

【請求項 12】

前記制御手段は、前記復号処理の処理速度が、前記第 1 の処理の処理速度よりも速い場合、前記復号処理の処理速度を低下させる

請求項 10 に記載の復号装置。

【請求項 13】

前記制御手段は、前記復号処理の処理速度が、前記第 2 の処理の処理速度よりも速い場合、前記復号処理の処理速度を低下させる

請求項 10 に記載の復号装置。

【請求項 14】

前記復号手段は、複数のコアを持つプロセッサによって実現され、

前記制御手段は、前記復号処理に割り当てる前記コアの数を増減させることにより、前記復号処理の処理速度を制御する

請求項 10 に記載の復号装置。

【請求項 15】

前記復号手段は、JPEG2000方式の可逆復号方式により前記復号処理を行う

請求項 10 に記載の復号装置。

【請求項 16】

前記制御手段は、前記コードストリームのファイルサイズの大きさに基づいて、前記復号手段による前記復号処理の処理速度を制御する

請求項 10 に記載の復号装置。

【請求項 17】

復号装置の第 1 の処理手段が、復号処理の対象である、画像データが符号化されて生成されたコードストリームに対して、前記復号処理の前に所定の第 1 の処理を行い、

前記復号装置の復号手段が、前記第 1 の処理が行われた前記コードストリームに対して前記復号処理を行い、前記画像データを生成し、

前記復号装置の第 2 の処理手段が、前記復号処理により前記コードストリームが復号されて生成された前記画像データに対して、所定の第 2 の処理を行い、

前記復号装置の制御手段が、前記第 1 の処理の処理速度および前記第 2 の処理の処理速度に応じて、前記復号処理の処理速度を制御する

復号方法。

【発明の詳細な説明】

【技術分野】

10

20

30

40

50

【 0 0 0 1 】

本発明は、符号化装置および方法、並びに、復号装置および方法に関し、特に、符号化処理または復号処理においてハードウェア資源をより有効に活用することができるようにした符号化装置および方法、並びに、復号装置および方法に関する。

【 背景技術 】

【 0 0 0 2 】

従来のアーカイブシステムまたは画像のデータベースでは、マスタ画像と呼ばれる非圧縮の画像が記憶・保持されており、そのマスタ画像が読み出されエンコードにより圧縮される。また、その圧縮結果のファイルは、ネットワーク等を介して配信されたり、記憶媒体に保存されたりする（例えば特許文献1および特許文献2参照）。

10

【 0 0 0 3 】

また、例えば、画像データが圧縮されたコードストリームがネットワークを介して受信され、デコードにより伸長される。伸長結果の画像データは、ネットワーク等を介して配信されたり、記憶媒体に保存されたりする。

【 先行技術文献 】

【 非特許文献 】

【 0 0 0 4 】

【 特許文献 1 】 特許 3 5 1 6 4 5 0 号 公 報

【 特許文献 2 】 特許 3 5 9 4 0 1 7 号 公 報

【 発明の概要 】

20

【 発明が解決しようとする課題 】

【 0 0 0 5 】

しかしながら、このようなシステムにおいて、エンコードやデコードの処理速度に対して、記憶媒体の読み出し速度や書き込み速度が十分に高速で無いことも考えられる。同様に、ネットワークの伝送速度がエンコードやデコードの処理速度に対して十分に高速で無いことも考えられる。

【 0 0 0 6 】

つまり、エンコードやデコードの処理速度に対して、それらの前段の処理部における処理速度、または、それらの後段の処理部における処理速度が十分に高速でないことが考えられる。換言すれば、例えば、エンコードやデコードの処理速度に対して、処理対象の画像データのエンコードへの入力速度や処理対象のコードストリームのデコードへの入力速度、または、処理結果のコードストリームのエンコードからの出力速度や処理結果の画像データのデコードからの出力速度が十分に高速でないことが考えられる。

30

【 0 0 0 7 】

このような場合、システム全体の処理速度が前段や後段の処理部における処理速度によって決定されるので、仮にエンコードやデコードの処理速度をより高速にしてもシステム全体の処理速度が向上しない、すなわち、不要に負荷を増大させてしまう恐れがあった。

【 0 0 0 8 】

また、逆に、エンコードやデコードの処理速度が、それらの前段の処理部における処理速度、および、それらの後段の処理部における処理速度に対して十分に高速でないことも考えられる。

40

【 0 0 0 9 】

このような場合、システム全体の処理速度がエンコードやデコードの処理速度によって決定されるので、仮にエンコードやデコードの処理速度をより高速にすることができるのであれば、不要にシステム全体の処理速度を低減させている恐れがあった。

【 0 0 1 0 】

以上のように、各処理部の処理速度の関係を考慮せずに、各処理部の処理速度を個別に調整しても、システム全体の処理速度が向上せずに、不要にハードウェア資源を消費してしまう恐れがあった。

【 0 0 1 1 】

50

特に、可逆圧縮の場合、その圧縮率が画像の難易度によって大きく変化するので、符号化や復号の処理速度が画像によって大きく変化する恐れがある。したがって、各処理部の処理速度の関係がより不安定になるので、より不要にハードウェア資源を消費してしまう恐れがあった。

【0012】

本発明は、このような状況に鑑みて提案されたものであり、符号化処理または復号処理においてハードウェア資源をより有効に活用することができるようにするものである。

【課題を解決するための手段】

【0013】

本発明の一側面は、符号化処理の対象である画像データに対して、前記符号化処理の前に所定の第1の処理を行う第1の処理手段と、前記第1の処理手段により前記第1の処理が行われた前記画像データに対して前記符号化処理を行い、コードストリームを生成する符号化手段と、前記符号化手段による前記符号化処理により前記画像データが符号化されて生成された前記コードストリームに対して、所定の第2の処理を行う第2の処理手段と、前記第1の処理手段による前記第1の処理の処理速度、および、前記第2の処理手段による前記第2の処理の処理速度に応じて、前記符号化手段による前記符号化処理の処理速度を制御する制御手段とを備える符号化装置である。

10

【0014】

前記制御手段は、前記符号化処理の処理速度が、前記第1の処理の処理速度よりも速い場合、前記符号化処理の処理速度を低下させることができる。

20

【0015】

前記制御手段は、前記符号化処理の処理速度が、前記第2の処理の処理速度よりも速い場合、前記符号化処理の処理速度を低下させることができる。

【0016】

前記制御手段は、前記符号化処理の処理速度が、前記第2の処理の処理速度よりも遅い場合、前記符号化処理の処理速度を上昇させることができる。

【0017】

前記符号化手段は、複数のコアを持つプロセッサによって実現され、前記制御手段は、前記符号化処理に割り当てる前記コアの数を増減させることにより、前記符号化処理の処理速度を制御することができる。

30

【0018】

前記符号化手段は、可逆符号化方式により前記符号化処理を行うことができる。

【0019】

前記符号化手段は、JPEG2000方式の可逆符号化方式により前記符号化処理を行い、前記制御手段は、前記画像データのコードブロック毎のゼロビットプレーン数を取得し、前記ゼロビットプレーン数から有効ビットプレーン数のピクチャ毎の総和を算出し、前記有効ビットプレーン数のピクチャ毎の総和を用いて、前記符号化手段による前記符号化処理の処理速度を制御することができる。

【0020】

前記制御手段による前記符号化処理の処理速度の制御に関する負荷制御情報を、前記コードストリームに付加する付加手段をさらに備えることができる。

40

【0021】

本発明の一側面は、また、符号化装置の第1の処理手段が、符号化処理の対象である画像データに対して、前記符号化処理の前に所定の第1の処理を行い、前記符号化装置の符号化手段が、前記第1の処理が行われた前記画像データに対して前記符号化処理を行い、コードストリームを生成し、前記符号化装置の第2の処理手段が、前記符号化処理により前記画像データが符号化されて生成された前記コードストリームに対して、所定の第2の処理を行い、前記符号化装置の制御手段が、前記第1の処理の処理速度および前記第2の処理の処理速度に応じて、前記符号化処理の処理速度を制御する符号化方法である。

【0022】

50

本発明の他の側面は、復号処理の対象である、画像データが符号化されて生成されたコードストリームに対して、前記復号処理の前に所定の第1の処理を行う第1の処理手段と、前記第1の処理手段により前記第1の処理が行われた前記コードストリームに対して前記復号処理を行い、前記画像データを生成する復号手段と、前記復号手段による前記復号処理により前記コードストリームが復号されて生成された前記画像データに対して、所定の第2の処理を行う第2の処理手段と、前記第1の処理手段による前記第1の処理の処理速度、および、前記第2の処理手段による前記第2の処理の処理速度に応じて、前記復号手段による前記復号処理の処理速度を制御する制御手段とを備える復号装置である。

【0023】

前記制御手段は、前記復号処理の処理速度が、前記第1の処理の処理速度よりも遅い場合、前記復号処理の処理速度を上昇させることができる。

10

【0024】

前記制御手段は、前記復号処理の処理速度が、前記第1の処理の処理速度よりも速い場合、前記復号処理の処理速度を低下させることができる。

【0025】

前記制御手段は、前記復号処理の処理速度が、前記第2の処理の処理速度よりも速い場合、前記復号処理の処理速度を低下させることができる。

【0026】

前記復号手段は、複数のコアを持つプロセッサによって実現され、前記制御手段は、前記復号処理に割り当てる前記コアの数を増減させることにより、前記復号処理の処理速度を制御することができる。

20

【0027】

前記復号手段は、複数のコアを持つプロセッサによって実現され、前記制御手段は、前記復号処理に割り当てる前記コアの数を増減させることにより、前記復号処理の処理速度を制御することができる。

【0028】

前記制御手段は、前記コードストリームのファイルサイズの大きさに基づいて、前記復号手段による前記復号処理の処理速度を制御することができる。

【0029】

本発明の他の側面は、また、復号装置の第1の処理手段が、復号処理の対象である、画像データが符号化されて生成されたコードストリームに対して、前記復号処理の前に所定の第1の処理を行い、前記復号装置の復号手段が、前記第1の処理が行われた前記コードストリームに対して前記復号処理を行い、前記画像データを生成し、前記復号装置の第2の処理手段が、前記復号処理により前記コードストリームが復号されて生成された前記画像データに対して、所定の第2の処理を行い、前記復号装置の制御手段が、前記第1の処理の処理速度および前記第2の処理の処理速度に応じて、前記復号処理の処理速度を制御する復号方法である。

30

【0030】

本発明の一側面においては、符号化処理の対象である画像データに対して、前記符号化処理の前に所定の第1の処理が行われ、前記第1の処理が行われた前記画像データに対して前記符号化処理が行われ、コードストリームが生成され、前記符号化処理により前記画像データが符号化されて生成された前記コードストリームに対して、所定の第2の処理が行われ、前記第1の処理の処理速度および前記第2の処理の処理速度に応じて、前記符号化処理の処理速度が制御される。

40

【0031】

本発明の他の側面においては、復号処理の対象である、画像データが符号化されて生成されたコードストリームに対して、復号処理の前に所定の第1の処理が行われ、第1の処理が行われたコードストリームに対して復号処理が行われ、画像データが生成され、復号処理によりコードストリームが復号されて生成された画像データに対して、所定の第2の処理が行われ、第1の処理の処理速度および第2の処理の処理速度に応じて、復号処理の

50

処理速度が制御される。

【発明の効果】

【0032】

本発明によれば、情報を処理することができる。特に、符号化処理または復号処理においてハードウェア資源をより有効に活用することができる。

【図面の簡単な説明】

【0033】

【図1】本発明を適用した符号化装置の主な構成例を示すブロック図である。

【図2】図1の符号化装置の具体的な例を示す図である。

【図3】図1の符号化装置の具体的な例を示す図である。

10

【図4】制御部の主な構成例を示すブロック図である。

【図5】符号化制御処理の流れの例を説明するフローチャートである。

【図6】負荷制御処理の流れの例を説明するフローチャートである。

【図7】画像符号化部の主な構成例を示すブロック図である。

【図8】サブバンドの例を説明する図である。

【図9】コードブロックの例を説明する図である。

【図10】ビットプレーンの例を説明する図である。

【図11】制御部の主な構成例を示すブロック図である。

【図12】ゼロビットプレーンの例を説明する図である。

【図13】有効ビットプレーン数から処理速度の補正量を求めるためのテーブル情報の例を説明する図である。

20

【図14】符号化制御処理の流れの例を説明するフローチャートである。

【図15】ハードウェア資源の例を説明するブロック図である。

【図16】制御部の主な構成例を示すブロック図である。

【図17】符号化制御処理の流れの例を説明するフローチャートである。

【図18】制御部の主な構成例を示すブロック図である。

【図19】メインヘッダの例を説明する模式図である。

【図20】コマンドシンタックスの構成例を説明する図である。

【図21】符号化制御処理の流れの例を説明するフローチャートである。

【図22】復号装置の主な構成例を示すブロック図である。

30

【図23】図1の符号化装置の具体的な例を示す図である。

【図24】図1の符号化装置の具体的な例を示す図である。

【図25】制御部の主な構成例を示すブロック図である。

【図26】復号制御処理の流れの例を説明するフローチャートである。

【図27】負荷制御処理の流れの例を説明するフローチャートである。

【図28】画像復号部の主な構成例を示すブロック図である。

【図29】制御部の主な構成例を示すブロック図である。

【図30】復号制御処理の流れの例を説明するフローチャートである。

【図31】制御部の主な構成例を示すブロック図である。

【図32】復号制御処理の流れの例を説明するフローチャートである。

40

【発明を実施するための形態】

【0034】

以下、発明を実施するための形態（以下実施の形態とする）について説明する。なお、説明は以下の順序で行う。

1. 第1の実施の形態（符号化制御処理）
2. 第2の実施の形態（符号化方式がJPEG2000の場合の符号化制御処理）
3. 第3の実施の形態（ハードウェア資源を割り当てる符号化制御処理）
4. 第4の実施の形態（復号制御処理に負荷情報を提供する符号化制御処理）
5. 第5の実施の形態（復号制御処理）
6. 第6の実施の形態（符号化方式がJPEG2000の場合の復号制御処理）

50

7. 第7の実施の形態（ハードウェア資源を割り当てる復号制御処理）

【0035】

< 1. 第1の実施の形態 >

[符号化装置の構成]

図1は、本発明を適用した符号化装置の主な構成例を示すブロック図である。図1に示される符号化装置100は、画像データを所定の符号化方式で符号化し、コードストリームを生成する装置である。符号化装置100は、制御部101、入力系111、画像符号化部112、および出力系113を有する。

【0036】

制御部101は、入力系111乃至出力系113を制御し、画像データの符号化に関する処理を実行させる。

【0037】

入力系111は、任意のハードウェア資源により実現される1つまたは複数の処理部よりなる系である。入力系111は、制御部101により制御され、符号化処理の対象となる画像データに対して、例えば、記憶媒体から画像データを読み出したり、ネットワークを介して他の装置より送信された画像データを受信したりする等、任意の処理を行う。また、入力系111は、制御部101により制御され、矢印121に示されるように、任意の処理を施したその画像データを画像符号化部112に供給する。

【0038】

画像符号化部112は、制御部101により制御され、矢印121に示されるように、入力系111より供給される画像データを取得する。また、画像符号化部112は、制御部101により制御され、取得した画像データを所定の方式で符号化してコードストリームを生成する。さらに、画像符号化部112は、制御部101により制御され、矢印122に示されるように、生成したコードストリームを出力系113に供給する。

【0039】

出力系113は、任意のハードウェア資源により実現される1つまたは複数の処理部よりなる系である。出力系113は、制御部101により制御され、矢印122に示されるように、画像符号化部112より供給されるコードストリームを取得する。また、出力系113は、制御部101により制御され、取得したコードストリームに対して、例えば記憶媒体に記憶したり、ネットワークを介して他の装置に送信したりする等、任意の処理を行う。

【0040】

制御部101は、矢印131に示されるように、入力系111から、入力系111が行う処理の処理速度を示す入力レート情報を取得する。同様に、制御部101は、矢印132に示されるように、出力系113から、出力系113が行う処理の処理速度を示す出力レート情報を取得する。さらに、制御部101は、矢印133に示されるように、画像符号化部112から、画像符号化部112が行う画像符号化処理の負荷量を示す負荷情報133を取得する。

【0041】

また、制御部101は、矢印134に示されるように入力系111に制御情報を供給して入力系111の処理を制御する。さらに、制御部101は、矢印135に示されるように画像符号化部112に制御情報を供給して画像符号化処理を制御する。また、制御部101は、矢印136に示されるよう出力系113に制御情報を供給して出力系113の処理を制御する。

【0042】

このとき、制御部101は、入力レート情報、出力レート情報、および負荷情報に基づいて、画像符号化部112が行う画像符号化処理の処理速度を決定する。

【0043】

ここで処理速度とは、単位時間当たりの処理量（所謂スループット）である。この処理速度は、例えば、単位時間当たりに処理される画像データ（コードストリーム）のデータ

10

20

30

40

50

量（所謂レート）により示される。つまり、入力レート情報は、入力系 1 1 1 において単位時間辺りに処理される画像データのデータ量（入力レート）を示す情報である。同様に、出力レート情報は、出力系 1 1 3 において単位時間辺りに処理される画像データのデータ量（出力レート）を示す情報である。

【 0 0 4 4 】

この入力レート情報は、入力系 1 1 1 に割り当てられたハードウェア資源の仕様であってもよいし、入力系 1 1 1 において実際に行われる処理の速度計測結果であってもよい。同様に、出力レート情報は、出力系 1 1 3 に割り当てられたハードウェア資源の仕様であってもよいし、出力系 1 1 3 において実際に行われる処理の速度計測結果であってもよい。

10

【 0 0 4 5 】

なお、入力レート情報および出力レート情報は、例えば単位時間当たりに処理されるデータ量のように、直接的にスループットを示す情報であってもよいが、例えばハードウェア資源の量のように、単位時間当たりの処理量に変換可能な、間接的にスループットを示す情報であってもよい。もちろん、入力レート情報および出力レート情報は、ハードウェア資源の仕様と速度計測結果を併用する等、複数の情報を含むようにしてもよい。

【 0 0 4 6 】

負荷情報は、画像符号化部 1 1 2 により行われる画像符号化処理の、予測される負荷の大きさを示す予測情報である。換言すれば、負荷情報は、画像符号化処理の処理速度（スループット）の予測値を示す。より具体的には、負荷情報は、例えば、画像データの符号化難易度を示す情報、過去の画像データのスループット、ハードウェア資源の割当量を示す情報、またはその他の設定情報等よりなる。つまり負荷情報は、画像符号化処理のスループットの予測値を求めることができる情報であればどのような情報であってもよい。

20

【 0 0 4 7 】

制御部 1 0 1 は、負荷情報より求められる画像符号化処理の処理速度を、入力系 1 1 1 の処理の処理速度や、出力系 1 1 3 の処理の処理速度と比較し、それらの比較結果に基づいて画像符号化処理の処理速度を制御する。

【 0 0 4 8 】

例えば、制御部 1 0 1 は、画像符号化処理の処理速度が入力系 1 1 1 の処理の処理速度より速い場合、画像符号化処理の処理速度を入力系 1 1 1 の処理速度まで低下させる。また、例えば、制御部 1 0 1 は、画像符号化処理の処理速度が出力系 1 1 3 の処理の処理速度より速い場合、画像符号化処理の処理速度を出力系 1 1 3 の処理速度まで低下させる。さらに、例えば、制御部 1 0 1 は、画像符号化処理の処理速度が出力系 1 1 3 の処理の処理速度より遅い場合、画像符号化処理の処理速度を出力系 1 1 3 の処理速度まで上昇させる。

30

【 0 0 4 9 】

このように、制御部 1 0 1 は、入力系 1 1 1 や出力系 1 1 3 等、画像符号化部 1 1 2 の前段や後段の処理部が行う処理の処理速度も考慮して画像符号化処理の処理速度を決定するので、符号化装置 1 0 0 全体の処理速度を向上させるように画像符号化処理の処理速度を制御することができる。

40

【 0 0 5 0 】

このような制御により符号化装置 1 0 0 は、不要な負荷の増大を抑制するように、符号化処理にハードウェア資源を割り当てることができる。つまり、符号化装置 1 0 0 は、符号化処理においてハードウェア資源をより有効に活用することができる。

【 0 0 5 1 】

次に、このような符号化装置 1 0 0 の構成の具体的な例について説明する。

【 0 0 5 2 】

図 2 は、図 1 の符号化装置の具体的な例を示す図である。符号化装置 1 0 0 A は、図 1 の符号化装置 1 0 0 の構成の具体的な一例を示す。図 2 に示されるように、符号化装置 1 0 0 A は、制御部 1 0 1 と画像符号化部 1 1 2 がバス 1 4 1 に接続される。バス 1 4 1 に

50

は、さらに、記憶部 1 4 2、キャッシュメモリ 1 4 3、および入出力インタフェース 1 4 4 が接続される。

【 0 0 5 3 】

バス 1 4 1 は、バス 1 4 1 に接続される処理部間で授受されるデータの伝送媒体である。例えば PCI バス (Peripheral Components Interconnect bus) 等により構成される。記憶部 1 4 2 は、データを記憶する記憶媒体である。例えば、ハードディスクやフラッシュメモリ等により構成される。

【 0 0 5 4 】

キャッシュメモリ 1 4 3 は、データを一時的に保持する記憶媒体である。例えば RAM (Random Access Memory) 等の、記憶部 1 4 2 の読み出し速度や、入出力インタフェースの 1 4 4 のスループットに比べて高速にデータの授受が可能な半導体メモリにより構成される。

10

【 0 0 5 5 】

入出力インタフェース 1 4 4 は、符号化装置 1 0 0 A のデータの授受を行う。入出力インタフェース 1 4 4 は、例えば、符号化装置 1 0 0 A の外部のネットワーク 1 5 1 より供給されるデータを取得し、そのデータを、バス 1 4 1 を介してキャッシュメモリ 1 4 3 等に供給する。また、入出力インタフェース 1 4 4 は、例えば、画像符号化部 1 1 2 等からバス 1 4 1 を介して供給されるデータを取得し、そのデータを、ネットワーク 1 5 1 を介して他の装置に供給する。例えば、HDMI (High-Definition Multimedia Interface)、HD-SDI (High Definition Serial Digital Interface)、Ethernet (登録商標) (LAN (Local Area Network) ボード) 等により構成される。

20

【 0 0 5 6 】

このような構成の符号化装置 1 0 0 A において、符号化対象の画像データは記憶部 1 4 2 に記憶されている。制御部 1 0 1 は、点線矢印 1 6 1 に示されるように、記憶部 1 4 2 からその画像データを読み出し、バス 1 4 1 を介してキャッシュメモリ 1 4 3 に供給させ、キャッシュメモリ 1 4 3 にその画像データを保持させる。

【 0 0 5 7 】

制御部 1 0 1 は、また、点線矢印 1 6 2 に示されるように、キャッシュメモリ 1 4 3 に保持されている画像データを読み出し、バス 1 4 1 を介して画像符号化部 1 1 2 に供給させ、符号化させる。

30

【 0 0 5 8 】

制御部 1 0 1 は、さらに、点線 1 6 3 に示されるように、画像データが符号化されたコードストリームを、バス 1 4 1 を介して入出力インタフェース 1 4 4 に供給させ、入出力インタフェース 1 4 4 から外部のネットワーク 1 5 1 に出力させる。

【 0 0 5 9 】

つまり、この符号化装置 1 0 0 A においては、一点鎖線で示されるように、バス 1 4 1、記憶部 1 4 2、およびキャッシュメモリ 1 4 3 により入力系 1 1 1 が形成され、二点鎖線で示されるように、バス 1 4 1 および入出力インタフェース 1 4 4 により出力系 1 1 3 が形成される。

【 0 0 6 0 】

入力系 1 1 1 において、一般的に、記憶部 1 4 2 からのデータの読み出しレートは、バス 1 4 1 の伝送レートやキャッシュメモリ 1 4 3 のデータ入出力レートより低い。従って、図 1 を参照して説明したように、制御部 1 0 1 は、画像符号化部 1 1 2 の画像符号化処理の処理速度 (スループット) が記憶部 1 4 2 のデータ読み出し速度より速い場合、画像符号化部 1 1 2 の画像符号化処理の処理速度を減速させ、記憶部 1 4 2 のデータ読み出し速度 (入力系 1 1 1 の処理速度) に合わせる。

40

【 0 0 6 1 】

出力系 1 1 3 において、一般的に、入出力インタフェース 1 4 4 のデータの入出力レート (スループット) は、バス 1 4 1 の伝送レートより低い。従って、図 1 を参照して説明したように、制御部 1 0 1 は、画像符号化部 1 1 2 の画像符号化処理の処理速度 (スル-

50

プット)が入出力インタフェース144のデータの入出力速度より速い場合、画像符号化部112の画像符号化処理の処理速度を減速させ、入出力インタフェース144のデータ入出力速度(出力系113の処理速度)に合わせる。

【0062】

また、図1を参照して説明したように、制御部101は、画像符号化部112の画像符号化処理の処理速度(スループット)が入出力インタフェース144のデータの入出力速度より遅い場合、画像符号化部112の画像符号化処理の処理速度を加速させ、入出力インタフェース144のデータ入出力速度(出力系113の処理速度)に合わせる。

【0063】

このように制御部101が符号化処理を制御することにより、符号化装置100Aは、各部の処理速度を適切に調整し、不要な待機時間の発生を抑制することができる。すなわち、符号化装置100Aは、ハードウェア資源をより有効に利用して符号化処理を行うことができる。

10

【0064】

図3は、図1の符号化装置100の他の具体的な例を示す図である。

【0065】

符号化装置100Bは、図1の符号化装置100の構成の、他の具体的な一例を示す。図3に示されるように、符号化装置100Bは、制御部101と画像符号化部112がバス141に接続される。バス141には、さらに、キャッシュメモリ143、および入出力インタフェース144が接続される。

20

【0066】

このような構成の符号化装置100Bにおいて、符号化対象の画像データはネットワーク151を介して他の装置より供給される。制御部101は、点線矢印171に示されるように、入出力インタフェース144を制御して、その画像データを取得し、バス141を介してキャッシュメモリ143に供給させ、キャッシュメモリ143にその画像データを保持させる。

【0067】

制御部101は、また、点線矢印172に示されるように、キャッシュメモリ143に保持されている画像データを読み出し、バス141を介して画像符号化部112に供給させ、符号化させる。

30

【0068】

制御部101は、さらに、点線173に示されるように、画像データが符号化されたコードストリームを、バス141を介して入出力インタフェース144に供給させ、入出力インタフェース144から外部のネットワーク151に出力させる。

【0069】

つまり、この符号化装置100Bにおいては、一点鎖線で示されるように、バス141、キャッシュメモリ143、および入出力インタフェース144により入力系111が形成され、二点鎖線で示されるように、バス141および入出力インタフェース144により出力系113が形成される。

【0070】

入力系111において、一般的に、入出力インタフェース144のデータの入出力レート(スループット)は、バス141の伝送レートやキャッシュメモリ143のデータ入出力レートより低い。従って、図1を参照して説明したように、制御部101は、画像符号化部112の画像符号化処理の処理速度(スループット)が入出力インタフェース144のデータの入出力速度より速い場合、画像符号化部112の画像符号化処理の処理速度を減速させ、入出力インタフェース144のデータ入出力速度(入力系111の処理速度)に合わせる。

40

【0071】

また、図2の場合と同様に、制御部101は、画像符号化部112の画像符号化処理の処理速度(スループット)が入出力インタフェース144のデータの入出力速度より速い

50

場合、画像符号化部 1 1 2 の画像符号化処理の処理速度を減速させ、入出力インタフェース 1 4 4 のデータ入出力速度（出力系 1 1 3 の処理速度）に合わせる。

【 0 0 7 2 】

さらに、図 2 の場合と同様に、制御部 1 0 1 は、画像符号化部 1 1 2 の画像符号化処理の処理速度（スループット）が入出力インタフェース 1 4 4 のデータの入出力速度より遅い場合、画像符号化部 1 1 2 の画像符号化処理の処理速度を加速させ、入出力インタフェース 1 4 4 のデータ入出力速度（出力系 1 1 3 の処理速度）に合わせる。

【 0 0 7 3 】

このように制御部 1 0 1 が符号化処理を制御することにより、符号化装置 1 0 0 B は、各部の処理速度を適切に調整し、不要な待機時間の発生を抑制することができる。すなわち、符号化装置 1 0 0 B は、ハードウェア資源をより有効に利用して符号化処理を行うことができる。

10

【 0 0 7 4 】

次に、このような制御を行う制御部 1 0 1 について説明する。図 4 は、制御部の主な構成例を示すブロック図である。図 4 に示されるように、制御部 1 0 1 は、入力レート情報取得部 1 8 1、出力レート情報取得部 1 8 2、符号化処理制御部 1 8 3、負荷情報取得部 1 8 4、負荷制御部 1 8 5、および記憶部 1 8 6 を有する。

【 0 0 7 5 】

入力レート情報取得部 1 8 1 は、入力系 1 1 1 を構成するデバイスより、入力系 1 1 1 のスループットを示す入力レート情報 1 9 1 を取得し、それを記憶部 1 8 6 に記憶させる。出力レート情報取得部 1 8 2 は、出力系 1 1 3 を構成するデバイスより、出力系 1 1 3 のスループットを示す出力レート情報 1 9 2 を取得し、それを記憶部 1 8 6 に記憶させる。

20

【 0 0 7 6 】

なお、この入力レート情報 1 9 1 および出力レート情報 1 9 2 は、入力系 1 1 1 または出力系 1 1 3 において計測された値であってもよいが、入力系 1 1 1 または出力系 1 1 3 を構成するデバイスの仕様であってもよい。したがって、入力系 1 1 1 および出力系 1 1 3 を構成するデバイスが予め定められており、それらのデバイスの仕様を入力レート情報 1 9 1 および出力レート情報 1 9 2 とする場合、入力レート情報 1 9 1 および出力レート情報 1 9 2 は、例えば工場出荷時等に予め記憶部 1 8 6 に記憶させておくようにしてもよい。その場合、入力レート情報取得部 1 8 1 および出力レート情報取得部 1 8 2 は、省略することができる。

30

【 0 0 7 7 】

符号化処理制御部 1 8 3 は、画像符号化部 1 1 2 による画像データの符号化処理を制御する。符号化処理制御部 1 8 3 は、負荷制御部 1 8 5 において決定された負荷量となるように、符号化処理のスループットを制御する。

【 0 0 7 8 】

負荷情報取得部 1 8 4 は、画像符号化部 1 1 2 の画像符号化処理の負荷の大きさ（負荷量）の予測値を示す負荷情報を取得する。例えば、負荷情報取得部 1 8 4 は、この負荷情報として、画像符号化処理のスループットの予測値を直接的または間接的に示すスループット情報 1 9 3 を取得する。スループット情報 1 9 3 の内容は任意であるが、例えば、画像データの符号化の難易度を示す情報や、過去の符号化処理の単位時間当たりの処理量等により構成される。負荷情報取得部 1 8 4 は、取得した負荷情報を負荷制御部 1 8 5 に供給する。

40

【 0 0 7 9 】

負荷制御部 1 8 5 は、記憶部 1 8 6 から入力レート情報 1 9 1 および出力レート情報 1 9 2 を読み出し、それらと、負荷情報取得部 1 8 4 から供給された負荷情報（スループット情報 1 9 3）とに基づいて、画像符号化部 1 1 2 による画像符号化処理の負荷量を決定する。負荷制御部 1 8 5 は、符号化処理制御部 1 8 3 を制御することにより、決定した負荷量となるように画像符号化処理のスループットを制御する。

50

【0080】

記憶部186は、例えばRAM等の半導体メモリにより構成される記憶媒体であり、入力レート情報191や出力レート情報192等の情報を記憶する。

【0081】

[処理の流れ]

次に、以上のような符号化装置100において実行される各種処理の流れについて説明する。最初に、制御部101による符号化制御処理の流れの例を、図5のフローチャートを参照して説明する。

【0082】

符号化制御処理が開始されると、制御部101の入力レート情報取得部181は、ステップS101において、入力レート情報を取得する。ステップS102において、出力レート情報取得部182は、出力レート情報を取得する。ステップS103において、符号化処理制御部183は、画像データを入力系111（例えばキャッシュメモリ143）から読み出し、画像符号化部112に入力させる。

10

【0083】

ステップS104において、符号化処理制御部183は、画像符号化部112を制御し、入力させた画像データを符号化する画像符号化処理を実行させる。ステップS105において、負荷情報取得部184は、その符号化処理についてスループット情報193を取得する。

【0084】

20

ステップS106において、負荷制御部185は、入力レートまたは出力レートのうち、少なくともいずれか一方に基づいて、ステップS104の制御によって実行された画像符号化処理のスループットを制御する。

【0085】

ステップS107において、符号化処理制御部183は、画像符号化処理により生成されたコードストリームを出力系113に出力し、処理させる。

【0086】

ステップS108において、符号化処理制御部183は、符号化制御処理を終了するかどうかを判定する。例えば未処理の画像データが入力系111（例えばキャッシュメモリ143）に存在する等して、符号化制御処理を終了しないと判定された場合、ステップS101に処理が戻り、それ以降の処理が繰り返される。

30

【0087】

このとき、入力レートが固定値の場合、ステップS101の処理は省略可能である。同様に、出力レートが固定値の場合、ステップS102の処理は省略可能である。

【0088】

ステップS108において、例えば未処理の画像データが入力系111（例えばキャッシュメモリ143）に存在せず、符号化制御処理を終了すると判定された場合、符号化制御処理が終了される。

【0089】

40

なお、ステップS105の処理（スループット情報の取得）は、その情報の内容に応じて適切なタイミングで行われる。したがって、例えば、ステップS104の処理より先にステップS105およびステップS106の処理が行われるようにしてもよい。また、例えば、画像符号化処理実行途中において、ステップS105およびステップS106の処理が行われるようにしてもよい。

【0090】

次に、図5のステップS106において実行される負荷制御処理の詳細な流れの例について、図6のフローチャートを参照して説明する。

【0091】

負荷制御処理が開始されると、負荷制御部185は、ステップS121において、入力レート情報191およびスループット情報193に基づいて、符号化処理のスループット

50

が入力レートより高いか否かを判定する。

【0092】

符号化処理のスループットが入力レートより高いと判定された場合、処理はステップS 1 2 2に進む。この場合、画像符号化処理の処理速度が速すぎて、入力系1 1 1からの画像データの供給が追いつかない。つまり、画像符号化処理がハードウェア資源を不要に消費している。そこで、負荷制御部1 8 5は、ステップS 1 2 2において、画像符号化部1 1 2の画像符号化処理の処理速度を減速させ、画像符号化処理のスループットを入力レートまで低下させる。ステップS 1 2 2の処理が終了すると、図5のステップS 1 0 6に戻り、それ以降の処理が実行される。

【0093】

符号化処理のスループットが入力レートより高くないと判定された場合、処理はステップS 1 2 3に進む。負荷制御部1 8 5は、ステップS 1 2 3において、出力レート情報1 9 2およびスループット情報1 9 3に基づいて、符号化処理のスループットが出力レートより高いか否かを判定する。

【0094】

符号化処理のスループットが出力レートより高いと判定された場合、処理はステップS 1 2 4に進む。この場合、画像符号化処理の処理速度が速すぎて、出力系1 1 3における処理がオーバフローする恐れがある。つまり、画像符号化処理がハードウェア資源を不要に消費している。そこで、負荷制御部1 8 5は、ステップS 1 2 4において、画像符号化部1 1 2の画像符号化処理の処理速度を減速させ、画像符号化処理のスループットを出力レートまで低下させる。ステップS 1 2 4の処理が終了すると、図5のステップS 1 0 6に戻り、それ以降の処理が実行される。

【0095】

符号化処理のスループットが出力レートより高くないと判定された場合、処理はステップS 1 2 5に進む。負荷制御部1 8 5は、ステップS 1 2 5において、出力レート情報1 9 2およびスループット情報1 9 3に基づいて、符号化処理のスループットが出力レートより低いかなかを判定する。

【0096】

符号化処理のスループットが出力レートより低いと判定された場合、処理はステップS 1 2 6に進む。この場合、画像符号化処理の処理速度が遅すぎて、出力系1 1 3において処理の無駄な空き時間（待機時間）が生じる。つまり、画像符号化処理のハードウェア資源の消費量が少なすぎる。ハードウェア資源に余裕があるのであれば、さらに画像符号化処理がハードウェア資源を消費するようにすることにより、符号化装置1 0 0のスループットを向上させることができる。そこで、負荷制御部1 8 5は、ステップS 1 2 6において、画像符号化部1 1 2の画像符号化処理の処理速度を加速させ、画像符号化処理のスループットを出力レートまで上昇させる。ステップS 1 2 6の処理が終了すると、図5のステップS 1 0 6に戻り、それ以降の処理が実行される。

【0097】

符号化処理のスループットが出力レートより低くないと判定された場合、負荷制御処理が終了され、図5のステップS 1 0 6に戻り、それ以降の処理が実行される。

【0098】

以上のような制御処理が行われることにより、符号化装置1 0 0は、符号化処理においてハードウェア資源をより有効に活用することができる。

【0099】

なお、図6においては、ステップS 1 2 1、ステップS 1 2 3、またはステップS 1 2 5の各判定処理を、図6に示されるフローチャートの順に行うように説明したが、各判定条件の優先順位は任意であり、どの判定から行うようにしてもよい。また、複数の判定を満たすときのみ符号化処理のスループットの制御（例えば、ステップS 1 2 2、ステップS 1 2 4、またはステップS 1 2 6）を行うようにしてもよい。

【0100】

< 2 . 第 2 の実施の形態 >

[画像符号化部の構成]

画像符号化部 1 1 2 の符号化方法は任意であり、例えば、JPEG (Joint Photographic Experts Group)、MPEG (Moving Picture Experts Group)、AVC (Advanced Video Coding)、JPEG-LS (Lossless JPEG)、または、JPEG2000 Losslessなどであってもよいし、これら以外の方法であっても良い。

【 0 1 0 1 】

以下に符号化方式がJPEG2000の場合について説明する。図 7 は、画像符号化部 1 1 2 の主な構成例を示すブロック図である。図 7 に示されるように、画像符号化部 1 1 2 は、DC レベルシフト部 2 0 1、ウェーブレット変換部 2 0 2、量子化部 2 0 3、コードブロック化部 2 0 4、およびビットプレーン展開部 2 0 5 を有する。

10

【 0 1 0 2 】

DC レベルシフト部 2 0 1 は、後段のウェーブレット変換を効率的に行うために、矢印 2 3 1 のように画像符号化部 1 1 2 に入力された画像データのDC成分のレベルシフトを行う。例えば、RGB信号が正の値 (符号無しの整数) を持っている。そこで、DCレベルシフト部 2 0 1 は、そのことを利用し、原信号のダイナミックレンジを半分にするレベルシフトを行うことで、圧縮効率の向上を図る。従ってYCbCr信号のCbやCr (色差信号) の様に符号 (正負両方あり) の整数値を持つ信号を原信号とする場合、このレベルシフトは行われない。

20

【 0 1 0 3 】

ウェーブレット変換部 2 0 2 は、通常低域フィルタと高域フィルタから構成されるフィルタバンクによって実現される。また、デジタルフィルタは通常複数タップ長のインパルス応答 (フィルタ係数) を有するので、ウェーブレット変換部 1 2 2 は、フィルタリングが行えるだけの入力画像を予めバッファリングするバッファを有する。

【 0 1 0 4 】

ウェーブレット変換部 2 0 2 は、矢印 2 3 2 のようにDCレベルシフト部 2 0 1 より出力された画像データを、フィルタリングに最低限必要なデータ量以上取得すると、そのDCレベルシフト後の画像データに対して、所定のウェーブレット変換フィルタを用いてフィルタリングを行い、ウェーブレット係数を生成する。なお、ウェーブレット変換部 2 0 2 は、画像の垂直方向および水平方向のそれぞれに対して、画像データを低域成分と高域成分に分離するフィルタリングを行う。

30

【 0 1 0 5 】

そして、ウェーブレット変換部 2 0 2 は、このようなフィルタリング処理を、図 8 に示されるように、垂直方向および水平方向の両方において低域成分として分離されたサブバンドに対して再帰的に所定回数繰り返す。これは、画像のエネルギーの多くが低域成分に集中しているからである。

【 0 1 0 6 】

図 8 は、分割レベル数 3 のウェーブレット変換処理により生成されるサブバンドの構成例を示す図である。この場合、ウェーブレット変換部 2 0 2 は、まず、画像全体をフィルタリングし、サブバンド 3LL (図示せず)、3HL、3LH、および 3HH を生成する。次に、ウェーブレット変換部 2 0 2 は、生成されたサブバンド 3LL に対して再度フィルタリングを行い、2LL (図示せず)、2HL、2LH、および 2HH を生成する。さらに、ウェーブレット変換部 1 2 2 は、生成されたサブバンド 2LL に対して再度フィルタリングを行い、0LL、1HL、1LH、および 1HH を生成する。

40

【 0 1 0 7 】

ウェーブレット変換部 2 0 2 は、フィルタリングにより得られたウェーブレット係数を、サブバンド毎に、矢印 2 3 3 に示されるように量子化部 2 0 3 に供給する。量子化部 2 0 3 は、供給されたウェーブレット係数を量子化する。この量子化の方法は任意であるが、量子化ステップサイズで除算するスカラ量子化が一般的である。量子化部 2 0 3 は、量子化により得られた量子化係数を、矢印 2 3 4 に示されるように、コードブロック化部 2

50

04に供給する。なお、これより後段においては、ウェーブレット係数の代わりに量子化係数が供給されることになるが、この量子化係数もウェーブレット係数の場合と基本的に同様に扱われる。したがって、以下においては、必要でない限りその点についての説明は省略し、単に係数または係数データと称する。

【0108】

なお、画像符号化部112が、復号処理により元のデータを完全に復元可能な可逆符号化方式により画像データを符号化する場合、この量子化部203の処理は省略され、矢印235に示されるように、ウェーブレット変換部202の出力がコードブロック化部204に供給される。

【0109】

ウェーブレット係数は、コードブロック化部204で、エントロピ符号化の処理単位である所定の大きさのコードブロックに分割される。図9は各サブバンド中のコードブロックの位置関係を示したものである。例えば64×64画素程度のサイズのコードブロックが、分割後のすべてのサブバンド中に生成される。図4の例において、最も分割レベルが小さい3HHのサブバンドの大きさが例えば640×320画素であるとする、64×64画素のコードブロックは合計50個存在することになる。後段の各処理部は、このコードブロック毎に処理を行う。

【0110】

コードブロック化部204は、矢印236に示されるように、各コードブロックをビットプレーン展開部205に供給する。ビットプレーン展開部205は、係数データを、ビットの位毎のビットプレーンに展開する。

【0111】

ビットプレーンは、所定の数のウェーブレット係数よりなる係数群を、1ビット毎、つまり位毎に分割（スライス）したものである。つまり、ビットプレーンは、その係数群の互いに同一の位のビット（係数ビット）の集合である。

【0112】

図10にその具体例を示す。図10の左図は縦4個、横4個の計16個の係数を示している。この16個の係数のうち、絶対値が最大のものは13で、2進数で1101と表現される。ビットプレーン展開部205は、このような係数群を、絶対値を示す4枚のビットプレーン（絶対値のビットプレーン）と、符号を示す1枚のビットプレーン（符号のビットプレーン）に展開する。つまり、図10中左の係数群は、図10中右に示されるように、4枚の絶対値のビットプレーンと1枚の符号のビットプレーンに展開される。ここで、絶対値のビットプレーンの要素はすべて0か1の値をとる。また、符号を示すビットプレーンの要素は、係数の値が正であることを示す値、係数の値が0であることを示す値、または係数の値がマイナスを示す値のいずれかをとる。

【0113】

画像符号化部112は、さらに、ビットモデリング部206、算術符号化部207、符号量加算部208、制御部209、ヘッダ生成部210、およびパケット生成部211を有する。

【0114】

ビットプレーン展開部205は、展開したビットプレーンを、矢印237に示されるように、ビットモデリング部206に供給する。

【0115】

ビットモデリング部206および算術符号化部207は、EBCOT（Embedded Coding with Optimized Truncation）部221として動作し、入力される係数データに対して、JPEG2000規格で定められたEBCOTと呼ばれるエントロピ符号化を行う。EBCOTは、所定の大きさのブロック毎にそのブロック内の係数の統計量を測定しながら符号化を行う手法である。

【0116】

ビットモデリング部206は、JPEG2000規格で定められた手順に従って、係数データに対してビットモデリングを行い、矢印238に示されるように算術符号化部207にコン

10

20

30

40

50

テキストを送出する。算術符号化部 2 0 7 は、係数のビットプレーンを算術符号化する。

【 0 1 1 7 】

コードブロックの縦横のサイズは 4 から 2 5 6 まで 2 のべき乗で、通常使用される大きさは、 $3 2 \times 3 2$ 、 $6 4 \times 6 4$ 、 $1 2 8 \times 3 2$ 等がある。係数値が n ビットの符号付き 2 進数で表されていて、bit 0 から bit ($n - 2$) が LSB から MSB までのそれぞれのビットを表すとする。残りの 1 ビットは符号を示す。符号ブロックの符号化は、MSB 側のビットプレーンから順番に、例えば Significant Propagation Pass、Magnitude Refinement Pass、および Cleanup Pass の 3 種類の符号化パスによって行われる。

【 0 1 1 8 】

算術符号化部 2 0 7 は、生成したコードストリームを、矢印 2 3 9 に示されるように、符号量加算部 2 0 8 に供給する。符号量加算部 2 0 8 は、そのコードストリームの符号量をカウントし、累積する。そして、符号量加算部 2 0 8 は、そのコードストリームを、矢印 2 4 2 および矢印 2 4 3 に示されるように、ヘッダ作成部 2 1 0 およびパケット生成部 2 1 1 に供給するとともに、矢印 2 4 0 に示されるように、符号量の累積値を制御部 2 0 9 に供給する。制御部 2 0 9 は、供給された符号量の累積値と目標符号量とを比較し、累積値が目標符号量より小さい場合、矢印 2 4 1 に示されるように、EBCOT 部 2 2 1 を制御し、次のビットプレーンの符号化を行わせる。EBCOT 部 2 2 1 は、その制御に従って次に重要なビットプレーンを符号化し、生成したコードストリームを符号量加算部 2 0 8 に供給する。符号量加算部 2 0 8 は、そのコードストリームの符号量をカウントして累積し、累積値を制御部 2 0 9 に供給する。

【 0 1 1 9 】

累積値が目標符号量に達するまで、以上のような処理が繰り返される。そして、累積値が目標符号量に達すると、制御部 2 0 9 は、EBCOT 部 2 2 1 を制御し、符号化処理を終了させる。

【 0 1 2 0 】

パケット生成部 2 1 1 は、供給された符号化コードストリームをパケット化する。ヘッダ生成部 2 1 0 は、そのパケットのヘッダ情報を生成し、そのヘッダ情報を矢印 2 4 4 に示されるように、パケット生成部 2 1 1 に供給する。パケット生成部 2 1 1 は、そのヘッダ情報を用いてパケット化を行う。生成されたパケットは矢印 2 4 5 に示されるように、画像符号化部 1 1 2 の外部に出力される。

【 0 1 2 1 】

[制御部の構成]

次に、このような画像符号化部 1 1 2 を制御する制御部について説明する。図 1 1 は、制御部の主な構成例を示すブロック図である。図 1 1 において、制御部 1 0 1 は、基本的に図 4 を参照して説明した場合と同様の構成を有する。すなわち、制御部 1 0 1 は、符号化処理制御部 1 8 3 乃至記憶部 1 8 6 を有する。

【 0 1 2 2 】

ただし、この場合、制御部 1 0 1 の負荷情報取得部 1 8 4 は、負荷情報として、画像符号化部 1 1 2 からゼロビットプレーン数 2 5 1 を取得する。

【 0 1 2 3 】

ゼロビットプレーンは、画像符号化部 1 1 2 のビットプレーン展開部 2 0 5 により展開された各ビットプレーンのうち、MSB から連続する、係数が全てゼロのビットプレーンであることを示す。つまり、ゼロビットプレーン数は、全ての係数がゼロであるビットプレーンの MSB からの連続数を示す。この値は、ビットプレーン展開部 2 0 5 がコードブロックをビットプレーンに展開した時点で決定される。負荷情報取得部 1 8 4 は、このように生成されるゼロビットプレーン数 2 5 1 を、画像符号化部 1 1 2 のビットプレーン展開部 2 0 5 より取得する。

【 0 1 2 4 】

図 1 2 は、ゼロビットプレーンの例を説明する図である。図 1 2 に示されるように、ゼロビットプレーン数 2 5 1 は、コードブロック毎に算出される。各コードブロックのピッ

10

20

30

40

50

ト深度Hからこのゼロビットプレーン数 (NUM_ZB) (図12中空白部分) を減算することにより、各コードブロックの有効ビットプレーン数 (NUM_BP) (図12中斜線部分) が算出される。この有効ビットプレーン数は、各コードブロックのおおよその情報量を示す。すなわち、各コードブロックのおおよその符号化の難易度を示す。

【0125】

記憶部186には、入力レート情報191や出力レート情報192の代わりに、テーブル情報252が記憶されている。テーブル情報252は、有効ビットプレーン数のピクチャ内の総和 (ALL_NUM_BP) から画像符号化処理の処理速度の制御量を決定するためのテーブル情報である。このテーブル情報252において、有効ビットプレーン数のピクチャ毎の総和と画像符号化処理の処理速度の制御量との対応関係は、入力レートと出力レートを考慮して作成されている。

10

【0126】

図13は、有効ビットプレーン数から処理速度の補正量を求めるためのテーブル情報の例を説明する図である。

【0127】

図13に示されるように、テーブル情報252においては、有効ビットプレーン数の総和 (ALL_NUM_BP) の取り得る値が複数の範囲に区切られ、各範囲に対して、画像符号化処理の処理速度の補正量が割り当てられている。例えば、図13の例の場合、有効ビットプレーン数の総和 (ALL_NUM_BP) が「501」乃至「1000」であるときを標準値とし、画像符号化処理の処理速度を変更しない (×1.0倍) とする。また、これより、有効ビット

20

【0128】

つまり、負荷制御部185は、負荷情報取得部184が取得したコードブロック毎のゼロビットプレーン数251から、各コードブロックの有効ビットプレーン数を求め、さらにその有効ビットプレーン数の総和をピクチャ毎に求める。負荷制御部185は、さらに、記憶部186より読み出したテーブル情報252を用いて、その有効ビットプレーン数の総和に対応する画像符号化処理の処理速度の制御量を求め、符号化処理制御部183を制御する。このようにすることにより、負荷制御部185は、入力レート、出力レート、および負荷情報に基づいて、画像符号化処理の負荷量を制御することになる。

30

【0129】

したがって、上述した場合と同様に、符号化装置100は、符号化処理においてハードウェア資源をより有効に活用することができる。

【0130】

なお、この場合、図4の入力レート情報取得部181および出力レート情報取得部182は、省略される。

【0131】

[処理の流れ]

次にこの場合の符号化制御処理の流れの例を図14のフローチャートを参照して説明する。

40

【0132】

符号化制御処理が開始されると、符号化処理制御部183は、ステップS201において、入力系111 (例えばキャッシュメモリ143) から画像データを画像符号化部112に入力させる。

【0133】

ステップS202において、符号化処理制御部183は、DCレベルシフト部201に、DCレベルをシフトさせる。ステップS203において、符号化処理制御部183は、ウェーブレット変換部202にウェーブレット変換させる。また、非可逆符号化方式の場合、符号化処理制御部183は、量子化部203にウェーブレット係数を量子化させる。ステ

50

ステップS 2 0 4において、符号化処理制御部 1 8 3は、コードブロック化部 2 0 4に、係数をコードブロック単位で分割させる。ステップS 2 0 5において、符号化処理制御部 1 8 3は、ビットプレーン展開部 2 0 5に、コードブロック毎の係数をビットプレーンに展開させる。

【 0 1 3 4 】

ステップS 2 0 6において、負荷情報取得部 1 8 4は、ビットプレーン展開部 2 0 5より各コードブロックのゼロビットプレーン数 2 5 1を取得する。負荷制御部 1 8 5は、ステップS 2 0 7において、ゼロビットプレーン数 2 5 1から、有効ビットプレーン数のピクチャ毎の総和を算出する。負荷制御部 1 8 5は、ステップS 2 0 8において、テーブル情報 2 5 2を参照し、有効ビットプレーン数のピクチャ毎の総和 (ALL_NUM_BP)に基づいて画像符号化処理のスループットを制御する。

10

【 0 1 3 5 】

ステップS 2 0 9において、符号化処理制御部 1 8 3は、EBCOT部 2 2 1等を制御し、負荷制御部 1 8 5により決定されたスループットでエントロピ符号化を行わせる。

【 0 1 3 6 】

符号化処理制御部 1 8 3は、ヘッダ生成部 2 1 0およびパケット生成部 2 1 1を制御し、ステップS 2 1 0において、パケットを生成させ、ステップS 2 1 1において、そのパケットを出力系 1 1 3に出力させる。

【 0 1 3 7 】

ステップS 2 1 2において、符号化処理制御部 1 8 3は、符号化制御処理を終了するかどうかを判定する。終了しないと判定された場合、ステップS 2 0 1に戻り、それ以降の処理が繰り返される。また、ステップS 2 1 2において、符号化制御処理を終了すると判定された場合、符号化制御処理が終了される。

20

【 0 1 3 8 】

このように制御が行われることにより、符号化装置 1 0 0は、符号化処理においてハードウェア資源をより有効に活用することができる。

【 0 1 3 9 】

なお、図 7 の画像符号化部 1 1 2において、EBCOT 2 2 1の処理の負荷が非常に大きく、DCレベルシフト部 2 0 1乃至ビットプレーン展開部 2 0 5の各処理は、それに比べて負荷が軽い。従って、ビットプレーン展開時に、その有効ビットプレーン数から画像符号化処理のスループットを求めることにより、制御部 1 0 1は、画像符号化処理をより高速に制御することができ、遅延時間の増大を抑制することができる。

30

【 0 1 4 0 】

また、特に、可逆符号化の場合、非可逆符号化に比べて、画像データの符号化の難易度による符号化処理の負荷の増減が大きい。したがって、ハードウェア資源を不要に消費する恐れが大きい。しかしながら、制御部 1 0 1が、上述したように制御を行うことにより、ピクチャ単位で画像符号化処理の負荷量を制御することができるので、可逆符号化の場合も、ハードウェア資源をより有効に活用することができる。

【 0 1 4 1 】

以上における画像符号化処理のスループットの制御方法は任意である。例えば、画像符号化処理のクロックの速度を変更するようによいし、画像符号化処理の実行に割り当てる時間を増減させてもよいし、符号化方法を変更するようによい。

40

【 0 1 4 2 】

また、例えば、画像符号化部 1 1 2の画像符号化処理をソフトウェアにより実現させるようにしてもよく、その場合、制御部 1 0 1が、画像符号化処理に割り当てるハードウェア資源の量を増減させることにより、画像符号化処理のスループットを制御するようによい。例えば、複数のCPU、複数のコアを有するCPU (Central Processing Unit)、またはマルチスレッド対応のCPUを用いて画像符号化処理を実行させる場合、制御部 1 0 1は、画像符号化処理に割り当てるCPU数、コア数、またはスレッド数を増減させることにより、画像符号化処理のスループットを制御するようによい。

50

【 0 1 4 3 】

< 3 . 第 3 の 実 施 の 形 態 >

[ハードウェア資源の構成]

以下に、複数コアを有するCPUを用いて画像符号化処理を行う場合のハードウェア資源割り当て方法の様子について説明する。図 1 5 は、このときのハードウェア資源の例を説明するブロック図である。

【 0 1 4 4 】

図 1 5 に示されるように、情報処理システム 3 0 0 は、画像符号化処理を実現する情報処理装置 3 0 1 を有する。その情報処理装置 3 0 1 には、PCIバス 3 0 2 を介して各種装置が接続される。例えば、記憶装置 3 0 3 や、複数台のビデオテープレコーダ (VTR) である VTR 3 0 4 - 1 乃至 VTR 3 0 4 - S が接続される。また、ユーザがこれらに対する操作入力を行うためのマウス 3 0 5、キーボード 3 0 6、並びに操作コントローラ 3 0 7 等も接続される。情報処理システム 3 0 0 は、これらの各装置により構成されるシステムであり、インストールされたプログラムによって、画像符号化処理や画像復号処理等を行うシステムである。

10

【 0 1 4 5 】

例えば情報処理システム 3 0 0 の情報処理装置 3 0 1 は、大容量の記憶装置 3 0 3 に記憶されている動画コンテンツを読み出して符号化し、得られた符号化データを記憶装置 3 0 3 に書き戻すことができるようになされている。さらに、情報処理装置 3 0 1 は、符号化処理により得られた符号化データを VTR 3 0 4 - 1 乃至 VTR 3 0 4 - S を介してビデオテープに記録したりすることができるようにもなされている。また、情報処理装置 3 0 1 は、VTR 3 0 4 - 1 乃至 VTR 3 0 4 - S に装着されたビデオテープに記録された動画コンテンツを記憶装置 3 0 3 に取り込み得るようにもなされている。その際、情報処理装置 3 0 1 が、動画コンテンツを符号化するようにしてもよい。

20

【 0 1 4 6 】

情報処理装置 3 0 1 は、マイクロプロセッサ 4 0 1、GPU (Graphics Processing Unit) 4 0 2 の他に、XDR (Extreme Data Rate) -RAM (Random Access Memory) 4 0 3 やサウスブリッジ 4 0 4 を有している。また、情報処理装置 3 0 1 は、HDD (Hard Disk Drive) 4 0 5、USB (Universal Serial Bus) インタフェース (USB I/F) 4 0 6、およびサウンド入出力コーデック 4 0 7 を有している。

30

【 0 1 4 7 】

GPU 4 0 2 は専用のバス 4 1 1 を介してマイクロプロセッサ 4 0 1 に接続される。XDR-RAM 4 0 3 は専用のバス 4 1 2 を介してマイクロプロセッサ 4 0 1 に接続される。サウスブリッジ 4 0 4 は、専用のバスを介してマイクロプロセッサ 4 0 1 の I/O コントローラ 4 4 4 に接続される。このサウスブリッジ 4 0 4 には、HDD 4 0 5、USB インタフェース 4 0 6、および、サウンド入出力コーデック 4 0 7 も接続されている。このサウンド入出力コーデック 4 0 7 にはスピーカ 4 2 1 が接続されている。また、GPU 4 0 2 にはディスプレイ 4 2 2 が接続される。サウスブリッジ 4 0 4 には、さらに、PCIバス 3 0 2 を介して、マウス 3 0 5、キーボード 3 0 6、VTR 3 0 4 - 1 乃至 VTR 3 0 4 - S、記憶装置 3 0 3、並びに、操作コントローラ 3 0 7 が接続される。

40

【 0 1 4 8 】

マウス 3 0 5 およびキーボード 3 0 6 は、ユーザの操作入力を受け、PCIバス 3 0 2 およびサウスブリッジ 4 0 4 を介して、ユーザの操作入力の内容を示す信号を、マイクロプロセッサ 4 0 1 に供給する。記憶装置 3 0 3 および VTR 3 0 4 - 1 乃至 VTR 3 0 4 - S は、所定のデータを記録または再生できるようになされている。PCIバス 3 0 2 にはさらに、必要に応じてドライブ 3 0 8 が接続され、磁気ディスク、光ディスク、光磁気ディスク、或いは半導体メモリなどのリムーバブルメディア 3 1 1 が適宜装着され、それらから読み出されたコンピュータプログラムが、必要に応じて HDD 4 0 5 にインストールされる。

【 0 1 4 9 】

マイクロプロセッサ 4 0 1 は、OS (Operating System) 等の基本プログラムを実行する

50

汎用のメインCPUコア 4 4 1 を有する。また、メインCPUコア 4 4 1 には、共有バス 4 4 5 を介して複数（この場合 8 個）のRISC（Reduced Instruction Set Computer）タイプの信号処理プロセッサである、サブCPUコア 4 4 2 - 1 乃至サブCPUコア 4 4 2 - 8 が接続される。また、共有バス 4 4 5 には、例えば 2 5 6 [MByte]の容量を持つXDR-RAM 4 0 3 に対するメモリコントロールを行うメモリコントローラ 4 4 3 も接続される。さらに、共有バス 4 4 5 には、サウスブリッジ 4 0 4 との間でデータの入出力を管理するI/O（In/Out）コントローラ 4 4 4 も接続される。マイクロプロセッサ 4 0 1 は、これらの構成が 1 チップに集積された、マルチコア構成のプロセッサである。マイクロプロセッサ 4 0 1 の動作周波数は、例えば 4 [GHz]である。

【 0 1 5 0 】

このマイクロプロセッサ 4 0 1 は、起動時、HDD 4 0 5 に格納された制御プログラムに基づき、HDD 4 0 5 に格納されている必要なアプリケーションプログラムを読み出してXDR-RAM 4 0 3 に展開し、この後このアプリケーションプログラム及びオペレータ操作に基づいて必要な制御処理を実行する。また、マイクロプロセッサ 4 0 1 は、ソフトウェアを実行することにより符号化処理や復号処理を実現する。例えば、マイクロプロセッサ 4 0 1 は、エンコードの結果得られた符号化ストリームを、サウスブリッジ 4 0 4 を介してHDD 4 0 5 に供給して記憶させることができる。また、マイクロプロセッサ 4 0 1 は、例えば、デコードした結果得られる動画コンテンツの再生映像を、GPU 4 0 2 へデータ転送し、ディスプレイ 4 2 2 に表示させることができる。

【 0 1 5 1 】

マイクロプロセッサ 4 0 1 内の各CPUコアの使用方法は任意であるが、例えば、メインCPUコア 4 4 1 が、制御部 1 0 1 の処理を行い、8 個のサブCPUコア 4 4 2 - 1 乃至サブCPUコア 4 4 2 - 8 がそれぞれ、画像符号化部 1 1 2 の画像符号化処理を行う。このような情報処理システム 3 0 0 のマイクロプロセッサ 4 0 1 における 8 個のサブCPUコア 4 4 2 - 1 乃至サブCPUコア 4 4 2 - 8 のそれぞれには、キャッシュメモリが内蔵されている。つまり、8 個のサブCPUコア 4 4 2 - 1 乃至サブCPUコア 4 4 2 - 8 は、それぞれ、各自の内部メモリであるキャッシュメモリと外部メモリであるXDR-RAM 4 0 3 の両方を利用することができる。

【 0 1 5 2 】

なお、以下において、サブCPUコア 4 4 2 - 1 乃至サブCPUコア 4 4 2 - 8 を互いに区別して説明する必要の無い場合、サブCPUコア 4 4 2 と称する。このサブCPUコア 4 4 2 の数は、図 1 5 においては 8 個設けられるように示されているが、実際には任意である。各サブCPUコア 4 4 2 には、事前に、メインCPUコア 4 4 1 により画像符号化処理を行うためのソフトウェアプログラムが割り当てられる（キャッシュメモリにロードされる）。各サブCPUコア 4 4 2 は、割り当てられたソフトウェアプログラムを実行することにより、符号化処理を実現する。

【 0 1 5 3 】

[制御部の構成]

図 1 6 は、図 1 5 に示されるようなハードウェア資源を用いて画像符号化処理を実行する場合の、制御部の主な構成例を示すブロック図である。

【 0 1 5 4 】

この場合の制御部 1 0 1 の構成も、基本的に図 4 を参照して説明した場合と同様であるが、負荷制御部 1 8 5 の代わりに、ハードウェア資源割り当て部 4 6 1 を有する。ハードウェア資源割り当て部 4 6 1 は、入力レート情報 1 9 1、出力レート情報 1 9 2、およびスループット情報 1 9 3 に基づいて、画像符号化処理に割り当てるハードウェア資源、すなわち、サブCPUコア 4 4 2 のコア数を制御する。符号化処理制御部 1 8 3 は、ハードウェア資源割り当て部 4 6 1 により割り当てられたサブCPUコア 4 4 2 を用いて画像符号化処理を行う。

【 0 1 5 5 】

なお、ここでは、入力レート情報 1 9 1 および出力レート情報 1 9 2 が予め記憶部 1 8

10

20

30

40

50

6に記憶されているものとし、入力レート情報取得部181および出力レート情報取得部182は省略されるものとする。もちろんこれらを有するようになってよい。

【0156】

[処理の流れ]

次に、この場合の符号化制御処理の流れの例について、図17のフローチャートを参照して説明する。

【0157】

符号化制御処理が開始されると、制御部101の符号化処理制御部183は、ステップS301において、画像データを入力系111（例えばキャッシュメモリ143）から読み出し、画像符号化部112に入力させる。

10

【0158】

ステップS302において、符号化処理制御部183は、画像符号化部112を制御し、入力させた画像データを符号化する画像符号化処理を実行させる。ステップS303において、負荷情報取得部184は、その符号化処理についてスループット情報193を取得する。

【0159】

ステップS304において、ハードウェア資源割り当て部461は、入力レートまたは出力レートのうち、少なくともいずれか一方に基づいて、画像符号化処理の一部または全部にハードウェア資源を割り当てる。この時、例えば符号化方法がJPEG2000の場合、符号化処理全体に割り当てるハードウェア資源を制御するようにしても良いし、EBCOT221により行われるエントロピ符号化処理に割り当てるハードウェア資源を制御するようにしてもよい。

20

【0160】

ステップS305において、符号化処理制御部183は、画像符号化処理により生成されたコードストリームを出力系113に出力し、処理させる。

【0161】

ステップS306において、符号化処理制御部183は、符号化制御処理を終了するかどうかを判定する。例えば未処理の画像データが入力系111（例えばキャッシュメモリ143）に存在する等して、符号化制御処理を終了しないと判定された場合、ステップS301に処理が戻り、それ以降の処理が繰り返される。

30

【0162】

ステップS306において、例えば未処理の画像データが入力系111（例えばキャッシュメモリ143）に存在せず、符号化制御処理を終了すると判定された場合、符号化制御処理が終了される。

【0163】

なお、ステップS303の処理（スループット情報の取得）は、その情報の内容に応じて適切なタイミングで行われる。したがって、例えば、ステップS302の処理より先にステップS303およびステップS304の処理が行われるようにしてもよい。また、例えば、画像符号化処理実行途中において、ステップS303およびステップS304の処理が行われるようにしてもよい。

40

【0164】

以上のような制御処理により、制御部101は、例えば、画像符号化処理のスループットが低すぎる場合、画像符号化処理に割り当てるハードウェア資源（例えばコア数）を増やすことにより、スループットを向上させることができる。また、例えば、画像符号化処理のスループットが高すぎる場合、制御部101は、画像符号化処理に割り当てるハードウェア資源（例えばコア数）を減らすことにより、スループットを低減させることができる。

【0165】

このように、制御部101は、入力レートや出力レート等に基づいて、画像符号化処理に割り当てるハードウェア資源の量をより適切な状態に保つことができる。画像符号化処

50

理に割り当てられないコアは、例えば他の処理に割り当てることができるので、以上のような制御により、サブCPUコア 4 4 2 がより有効に活用されるようになる。つまり、符号化装置 1 0 0 は、符号化処理においてハードウェア資源をより有効に活用することができる。

【 0 1 6 6 】

なお、画像符号化処理に割り当てるCPUの数を制御する場合や、画像符号化処理に割り当てるスレッド数を制御する場合も、基本的にコア数を制御する場合と同様に制御される。

【 0 1 6 7 】

< 4 . 第 4 の実施の形態 >

10

[制御部の構成]

なお、以上に説明した負荷量の制御に関する情報は、復号処理の際に、ハードウェア資源の割り当て制御に利用することができる。この情報としては、例えば、画像符号化処理の圧縮率、画像符号化処理のスループット、画像符号化処理に割り当てたコア数、または、それらを変更したときの入力レートや出力レート等のシステム環境条件等がある。もちろん、これら以外の情報であっても良い。

【 0 1 6 8 】

図 1 8 は、この場合の制御部の主な構成例を示すブロック図である。図 1 8 に示されるように、この場合、制御部 1 0 1 は、基本的に図 1 6 を参照して説明した場合と同様の構成を有するが、さらに情報付加部 4 7 1 を有する。

20

【 0 1 6 9 】

情報付加部 4 7 1 は、例えば、画像符号化処理の圧縮率、画像符号化処理のスループット、画像符号化処理に割り当てたコア数、または、それらを変更したときの入力レートや出力レート等のシステム環境条件等の負荷量の制御に関する情報を、負荷制御情報として、画像符号化処理により生成されたコードストリームのヘッダに付加する。情報付加部 4 7 1 は、例えば、コードストリームのメインヘッダに付加する。

【 0 1 7 0 】

図 1 9 は、符号化方式がJPEG2000の場合のコードストリームのメインヘッダの例を説明する模式図である。図 1 9 に示されるように、符号化方式がJPEG2000の場合、コードストリームのメインヘッダ 4 8 0 には、SOC 4 8 1 に続いてSIZ 4 8 2 が形成される。SIZ 4 8 2 には、COD 4 8 3、COC 4 8 4、QCD 4 8 5、QCC 4 8 6、RGN 4 8 7、POC 4 8 8、およびCOM 4 8 9 等のマーカセグメントが形成される。

30

【 0 1 7 1 】

JPEG2000においては、COM 4 8 9 のシンタックスは、図 2 0 に示されるように定められており、COM 4 9 1、Lcom 4 9 2、およびRcom 4 9 3 等のマーカが含まれる。Rcom 4 9 3 は、16ビットのパラメータであり、値「0」と「1」は予約されているが、それ以外の値は更改されており、自由に使用することができる。

【 0 1 7 2 】

情報付加部 4 7 1 は、このRcom 4 9 3 に、「0」と「1」以外の値を用いて、圧縮率等の付加制御情報を付加する。つまり、値「2」から値「65,535」までの値を定義可能なので、例えば、圧縮率 = 3分の2 = 0.6667 x 65,535 = 43,690 とすれば良い。

40

【 0 1 7 3 】

[処理の流れ]

この場合の符号化制御処理の流れの例を図 2 1 のフローチャートを参照して説明する。

【 0 1 7 4 】

符号化制御処理が開始されると、符号化処理制御部 1 8 3 は、ステップ S 4 0 1 において、画像データを入力系 1 1 1 (例えばキャッシュメモリ 1 4 3) から読み出し、画像符号化部 1 1 2 に入力させる。

【 0 1 7 5 】

ステップ S 4 0 2 において、符号化処理制御部 1 8 3 は、画像符号化部 1 1 2 を制御し

50

、入力させた画像データを符号化する画像符号化処理を実行させる。ステップS 4 0 3において、負荷情報取得部 1 8 4 は、その符号化処理についてスループット情報 1 9 3 を取得する。

【 0 1 7 6 】

ステップS 4 0 4において、ハードウェア資源割り当て部 4 6 1 は、入力レートまたは出力レートのうち、少なくともいずれか一方に基づいて、画像符号化処理の一部または全部にハードウェア資源を割り当てる。この時、例えば符号化方法がJPEG2000の場合、符号化処理全体に割り当てるハードウェア資源を制御するようにしても良いし、EBCOT 2 2 1 により行われるエントロピ符号化処理に割り当てるハードウェア資源を制御するようにしてもよい。

10

【 0 1 7 7 】

ステップS 4 0 5において、情報負荷部 4 7 1 は、例えば画像符号化処理の圧縮率等の、負荷制御情報をコードストリームのヘッダに登録する。

【 0 1 7 8 】

ステップS 4 0 6において、符号化処理制御部 1 8 3 は、画像符号化処理により生成されたコードストリームを出力系 1 1 3 に出力し、処理させる。

【 0 1 7 9 】

ステップS 4 0 7において、符号化処理制御部 1 8 3 は、符号化制御処理を終了するかどうかを判定する。例えば未処理の画像データが入力系 1 1 1 (例えばキャッシュメモリ 1 4 3) に存在する等して、符号化制御処理を終了しないと判定された場合、ステップS 4 0 1 に処理が戻り、それ以降の処理が繰り返される。

20

【 0 1 8 0 】

ステップS 4 0 7において、例えば未処理の画像データが入力系 1 1 1 (例えばキャッシュメモリ 1 4 3) に存在せず、符号化制御処理を終了すると判定された場合、符号化制御処理が終了される。

【 0 1 8 1 】

なお、ステップS 4 0 3の処理(スループット情報の取得)は、その情報の内容に応じて適切なタイミングで行われる。したがって、例えば、ステップS 4 0 2の処理より先にステップS 4 0 3およびステップS 4 0 4の処理が行われるようにしてもよい。また、例えば、画像符号化処理実行途中において、ステップS 4 0 3およびステップS 4 0 4の処理が行われるようにしてもよい。

30

【 0 1 8 2 】

以上のような制御処理により、制御部 1 0 1 は、例えば、画像の圧縮率、画像符号化処理に割り当てられたコア数、または、スループットを変更したときのシステム環境情報(変更条件の履歴情報)等を含む負荷制御情報を、復号装置に提供することができる。

【 0 1 8 3 】

< 5 . 第 5 の実施の形態 >

[復号装置の構成]

以上のような負荷制御は復号処理においても適用可能である。

【 0 1 8 4 】

図 2 2 は、復号装置の主な構成例を示すブロック図である。

40

【 0 1 8 5 】

図 2 2 に示される復号装置 6 0 0 は、符号化装置 1 0 0 の画像符号化処理に対応する方法で画像復号処理を行い、符号化装置 1 0 0 が生成したコードストリームを復号して画像データを得る装置である。復号装置 6 0 0 は、制御部 6 0 1、入力系 6 1 1、画像復号部 6 1 2、および出力系 6 1 3 を有する。

【 0 1 8 6 】

制御部 6 0 1 は、入力系 6 1 1 乃至出力系 6 1 3 を制御し、画像データの復号に関する処理を実行させる。

【 0 1 8 7 】

50

入力系 6 1 1 は、入力系 1 1 1 と同様に、任意のハードウェア資源により実現される 1 つまたは複数の処理部よりなる系である。入力系 6 1 1 は、制御部 6 0 1 により制御され、復号処理の対象となるコードストリームに対して、例えば、記憶媒体からコードストリームを読み出したり、ネットワークを介して他の装置より送信されたコードストリームを受信したりする等、任意の処理を行う。また、入力系 6 1 1 は、制御部 6 0 1 により制御され、矢印 6 2 1 に示されるように、任意の処理を施したその画像データを画像復号部 6 1 2 に供給する。

【 0 1 8 8 】

画像復号部 6 1 2 は、制御部 6 0 1 により制御され、矢印 6 2 1 に示されるように、入力系 6 1 1 より供給されるコードストリームを取得する。また、画像復号部 6 1 2 は、制御部 6 0 1 により制御され、取得したコードストリームを所定の方式で復号して画像データを生成する。さらに、画像復号部 6 1 2 は、制御部 6 0 1 により制御され、矢印 6 2 2 に示されるように、生成した画像データを出力系 6 1 3 に供給する。

【 0 1 8 9 】

出力系 6 1 3 は、出力系 1 1 3 と同様に、任意のハードウェア資源により実現される 1 つまたは複数の処理部よりなる系である。出力系 6 1 3 は、制御部 6 0 1 により制御され、矢印 6 2 2 に示されるように、画像復号部 6 1 2 より供給される画像データを取得する。また、出力系 6 1 3 は、制御部 6 0 1 により制御され、取得したコードストリームに対して、例えば記憶媒体に記憶したり、ネットワークを介して他の装置に送信したりする等、任意の処理を行う。

【 0 1 9 0 】

制御部 6 0 1 は、矢印 6 3 1 に示されるように、入力系 6 1 1 から、入力系 6 1 1 が行う処理の処理速度を示す入力レート情報を取得する。同様に、制御部 6 0 1 は、矢印 6 3 2 に示されるように、出力系 6 1 3 から、出力系 6 1 3 が行う処理の処理速度を示す出力レート情報を取得する。さらに、制御部 6 0 1 は、矢印 6 3 3 に示されるように、画像復号部 6 1 2 から、画像復号部 6 1 2 が行う画像復号処理の負荷量を示す負荷情報 6 3 3 を取得する。

【 0 1 9 1 】

また、制御部 6 0 1 は、矢印 6 3 4 に示されるように入力系 6 1 1 に制御情報を供給して入力系 6 1 1 の処理を制御する。さらに、制御部 6 0 1 は、矢印 6 3 5 に示されるように画像復号部 6 1 2 に制御情報を供給して画像復号処理を制御する。また、制御部 6 0 1 は、矢印 6 3 6 に示されるように出力系 6 1 3 に制御情報を供給して出力系 6 1 3 の処理を制御する。

【 0 1 9 2 】

このとき、制御部 6 0 1 は、入力レート情報、出力レート情報、および負荷情報に基づいて、画像復号部 6 1 2 が行う画像復号処理の処理速度を決定する。

【 0 1 9 3 】

なお、復号装置 6 0 0 の場合、負荷情報は、画像復号部 6 1 2 により行われる画像復号処理の、予測される負荷の大きさを示す予測情報である。基本的に符号化装置 1 0 0 の場合と同様の情報である。

【 0 1 9 4 】

制御部 6 0 1 は、負荷情報より求められる画像復号処理の処理速度を、入力系 6 1 1 の処理の処理速度や、出力系 6 1 3 の処理の処理速度と比較し、それらの比較結果に基づいて画像復号処理の処理速度を制御する。

【 0 1 9 5 】

例えば、制御部 6 0 1 は、画像復号処理の処理速度が入力系 6 1 1 の処理の処理速度より遅い場合、画像復号処理の処理速度を入力系 6 1 1 の処理速度まで上昇させる。また、例えば、制御部 6 0 1 は、画像復号処理の処理速度が入力系 6 1 1 の処理の処理速度より速い場合、画像復号処理の処理速度を入力系 6 1 1 の処理速度まで低下させる。さらに、例えば、制御部 6 0 1 は、画像復号処理の処理速度が出力系 6 1 3 の処理の処理速度より

10

20

30

40

50

速い場合、画像復号処理の処理速度を出力系 6 1 3 の処理速度まで低下させる。

【 0 1 9 6 】

このように、制御部 6 0 1 は、入力系 6 1 1 や出力系 6 1 3 等、画像復号部 6 1 2 の前段や後段の処理部が行う処理の処理速度も考慮して画像復号処理の処理速度を決定するので、復号装置 6 0 0 全体の処理速度を向上させるように画像復号処理の処理速度を制御することができる。

【 0 1 9 7 】

このような制御により復号装置 6 0 0 は、不要な負荷の増大を抑制するように、復号処理にハードウェア資源を割り当てることができる。つまり、復号装置 6 0 0 は、復号処理においてハードウェア資源をより有効に活用することができる。

10

【 0 1 9 8 】

次に、このような復号装置 6 0 0 の構成の具体的な例について説明する。

【 0 1 9 9 】

図 2 3 は、図 2 2 の復号装置の具体的な例を示す図である。復号装置 6 0 0 A は、図 2 2 の復号装置 6 0 0 の構成の具体的な一例を示す。図 2 3 に示されるように、復号装置 6 0 0 A は、制御部 6 0 1 と画像復号部 6 1 2 がバス 6 4 1 に接続される。バス 6 4 1 には、さらに、入出力インタフェース 6 4 1、キャッシュメモリ 6 4 3、および記憶部 6 4 4 が接続される。

【 0 2 0 0 】

バス 6 4 1 は、バス 1 4 1 と同様の伝送媒体であり、例えば PCI バス等により構成される。入出力インタフェース 6 4 2 は、入出力インタフェース 1 4 4 と同様の、外部のネットワーク 6 5 1 と接続される通信インタフェースであり、例えば、HDMI、HD-SDI、Ethernet (登録商標) (LAN ボード) 等により構成される。

20

【 0 2 0 1 】

キャッシュメモリ 6 4 3 は、キャッシュメモリ 1 4 3 と同様の記憶媒体であり、例えば RAM 等の、記憶部 6 4 4 の読み出し速度や、入出力インタフェースの 6 4 2 のスループットに比べて高速にデータの授受が可能な半導体メモリにより構成される。記憶部 6 4 4 は、記憶部 1 4 2 と同様の記憶媒体であり、例えば、ハードディスクやフラッシュメモリ等により構成される。

【 0 2 0 2 】

このような構成の復号装置 6 0 0 A において、復号対象のコードストリームは例えば、ネットワーク 6 5 1 を介して他の装置より供給される。制御部 6 0 1 は、点線矢印 6 6 1 に示されるように、入出力インタフェース 6 4 2 を制御して、そのコードストリームを取得し、バス 6 4 1 を介してキャッシュメモリ 6 4 3 に供給させ、キャッシュメモリ 6 4 3 にそのコードストリームを保持させる。

30

【 0 2 0 3 】

制御部 6 0 1 は、また、点線矢印 6 6 2 に示されるように、キャッシュメモリ 6 4 3 に保持されているコードストリームを読み出し、バス 6 4 1 を介して画像復号部 6 1 2 に供給させ、復号させる。

【 0 2 0 4 】

制御部 6 0 1 は、さらに、点線 6 6 3 に示されるように、コードストリームが符号化された画像データを、バス 1 4 1 を介して記憶部 6 4 4 に供給させ、記憶させる。

40

【 0 2 0 5 】

つまり、この復号装置 6 0 0 A においては、一点鎖線で示されるように、バス 6 4 1、入出力インタフェース 6 4 2、およびキャッシュメモリ 6 4 3 により入力系 6 1 1 が形成され、二点鎖線で示されるように、バス 6 4 1 および記憶部 6 4 4 により出力系 6 1 3 が形成される。

【 0 2 0 6 】

入力系 6 1 1 において、一般的に、入出力インタフェース 6 4 2 のデータの入出力レート (スループット) は、バス 6 4 1 の伝送レートやキャッシュメモリ 1 4 3 のデータ入出

50

レートより低い。従って、図 2 2 を参照して説明したように、制御部 6 0 1 は、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が入出力インタフェース 6 4 2 のデータの入出力速度より速い場合、画像復号部 6 1 2 の画像復号処理の処理速度を減速させ、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が入出力インタフェース 6 4 2 のデータの入出力速度より遅い場合、画像復号部 6 1 2 の画像復号処理の処理速度を加速させ、入出力インタフェース 6 4 2 のデータの入出力速度（入力系 6 1 1 の処理速度）に合わせる。

【 0 2 0 7 】

また、出力系 6 1 3 において、一般的に、記憶部 6 4 4 へのデータの書き込みレートは、バス 6 4 1 の伝送レートより低い。従って、図 2 2 を参照して説明したように、制御部 6 0 1 は、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が記憶部 6 4 2 のデータ書き込み速度より速い場合、画像復号部 6 1 2 の画像復号処理の処理速度を減速させ、記憶部 6 4 2 のデータ書き込み速度（入力系 6 1 1 の処理速度）に合わせる。

10

【 0 2 0 8 】

このように制御部 6 0 1 が復号処理を制御することにより、復号装置 6 0 0 A は、各部の処理速度を適切に調整し、不要な待機時間の発生を抑制することができる。すなわち、復号装置 6 0 0 A は、ハードウェア資源をより有効に利用して復号処理を行うことができる。

【 0 2 0 9 】

図 2 4 は、図 2 2 の復号装置 6 0 0 の他の具体的な例を示す図である。

20

【 0 2 1 0 】

復号装置 6 0 0 B は、図 2 2 の復号装置 6 0 0 の構成の、他の具体的な一例を示す。図 2 4 に示されるように、復号装置 6 0 0 B は、制御部 6 0 1 と画像復号部 6 1 2 がバス 6 4 1 に接続される。バス 6 4 1 には、さらに、キャッシュメモリ 6 4 3、および入出力インタフェース 6 4 2 が接続される。

【 0 2 1 1 】

このような構成の復号装置 6 0 0 B において、復号対象のコードストリームはネットワーク 6 5 1 を介して他の装置より供給される。制御部 6 0 1 は、点線矢印 6 7 1 に示されるように、入出力インタフェース 6 4 2 を制御して、そのコードストリームを取得し、バス 6 4 1 を介してキャッシュメモリ 6 4 3 に供給させ、キャッシュメモリ 6 4 3 にそのコードストリームを保持させる。

30

【 0 2 1 2 】

制御部 6 0 1 は、また、点線矢印 6 7 2 に示されるように、キャッシュメモリ 6 4 3 に保持されているコードストリームを読み出し、バス 6 4 1 を介して画像復号部 6 1 2 に供給させ、復号させる。

【 0 2 1 3 】

制御部 6 0 1 は、さらに、点線 6 7 3 に示されるように、コードストリームが復号された画像データを、バス 6 4 1 を介して入出力インタフェース 6 4 2 に供給させ、入出力インタフェース 6 4 2 から外部のネットワーク 6 5 1 に出力させる。

【 0 2 1 4 】

つまり、この復号装置 6 0 0 B においては、一点鎖線で示されるように、バス 6 4 1、キャッシュメモリ 6 4 3、および入出力インタフェース 6 4 2 により入力系 6 1 1 が形成され、二点鎖線で示されるように、バス 6 4 1 および入出力インタフェース 6 4 2 により出力系 6 1 3 が形成される。

40

【 0 2 1 5 】

入力系 6 1 1 において、一般的に、入出力インタフェース 6 4 2 のデータの入出力レート（スループット）は、バス 6 4 1 の伝送レートやキャッシュメモリ 6 4 3 のデータ入出力レートより低い。従って、図 2 2 を参照して説明したように、制御部 6 0 1 は、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が入出力インタフェース 6 4 2 のデータの入出力速度より遅い場合、画像復号部 6 1 2 の画像復号処理の処理速度を加速さ

50

せ、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が入出力インタフェース 6 4 2 のデータの入出力速度より速い場合、画像復号部 6 1 2 の画像復号処理の処理速度を減速させ、入出力インタフェース 6 4 2 のデータ入出力速度（入力系 6 1 1 の処理速度）に合わせる。

【 0 2 1 6 】

また、出力系 6 1 3 において、図 2 3 の場合と同様に、制御部 6 0 1 は、画像復号部 6 1 2 の画像復号処理の処理速度（スループット）が入出力インタフェース 6 4 2 のデータの入出力速度より速い場合、画像復号部 6 1 2 の画像復号処理の処理速度を減速させ、入出力インタフェース 6 4 2 のデータ入出力速度（出力系 6 1 3 の処理速度）に合わせる。

【 0 2 1 7 】

このように制御部 6 0 1 が復号処理を制御することにより、復号装置 6 0 0 B は、各部の処理速度を適切に調整し、不要な待機時間の発生を抑制することができる。すなわち、復号装置 6 0 0 B は、ハードウェア資源をより有効に利用して復号処理を行うことができる。

【 0 2 1 8 】

次に、このような制御を行う制御部 6 0 1 について説明する。図 2 5 は、制御部の主な構成例を示すブロック図である。図 2 5 に示されるように、制御部 6 0 1 は、入力レート情報取得部 6 8 1、出力レート情報取得部 6 8 2、復号処理制御部 6 8 3、負荷情報取得部 6 8 4、負荷制御部 6 8 5、および記憶部 6 8 6 を有する。

【 0 2 1 9 】

入力レート情報取得部 6 8 1 は、入力系 6 1 1 を構成するデバイスより、入力系 6 1 1 のスループットを示す入力レート情報 6 9 1 を取得し、それを記憶部 6 8 6 に記憶させる。出力レート情報取得部 6 8 2 は、出力系 6 1 3 を構成するデバイスより、出力系 6 1 3 のスループットを示す出力レート情報 6 9 2 を取得し、それを記憶部 6 8 6 に記憶させる。

【 0 2 2 0 】

なお、この入力レート情報 6 9 1 および出力レート情報 6 9 2 は、入力系 6 1 1 または出力系 6 1 3 において計測された値であってもよいが、入力系 6 1 1 または出力系 6 1 3 を構成するデバイスの仕様であってもよい。したがって、入力系 6 1 1 および出力系 6 1 3 を構成するデバイスが予め定められており、それらのデバイスの仕様を入力レート情報 6 9 1 および出力レート情報 6 9 2 とする場合、入力レート情報 6 9 1 および出力レート情報 6 9 2 は、例えば工場出荷時等に予め記憶部 6 8 6 に記憶させておくようにしてもよい。その場合、入力レート情報取得部 6 8 1 および出力レート情報取得部 6 8 2 は、省略することができる。

【 0 2 2 1 】

復号処理制御部 6 8 3 は、画像復号部 6 1 2 による画像データの復号処理を制御する。復号処理制御部 1 8 3 は、負荷制御部 6 8 5 において決定された負荷量となるように、復号処理のスループットを制御する。

【 0 2 2 2 】

負荷情報取得部 6 8 4 は、画像復号部 6 1 2 の画像復号処理の負荷の大きさ（負荷量）の予測値を示す負荷情報を取得する。例えば、負荷情報取得部 6 8 4 は、この負荷情報として、画像復号処理のスループットの予測値を直接的または間接的に示すスループット情報 6 9 3 を取得する。スループット情報 6 9 3 の内容は任意であるが、例えば、画像データの符号化の難易度を示す情報、過去の復号処理の単位時間当たりの処理量、またはコードストリームのファイルサイズ等により構成される。負荷情報取得部 6 8 4 は、取得した負荷情報を負荷制御部 6 8 5 に供給する。

【 0 2 2 3 】

負荷制御部 6 8 5 は、記憶部 6 8 6 から入力レート情報 6 9 1 および出力レート情報 6 9 2 を読み出し、それらと、負荷情報取得部 6 8 4 から供給された負荷情報（スループット情報 6 9 3）とに基づいて、画像復号部 6 1 2 による画像復号処理の負荷量を決定する

10

20

30

40

50

。負荷制御部 685 は、復号処理制御部 683 を制御することにより、決定した負荷量となるように画像復号処理のスループットを制御する。

【0224】

記憶部 686 は、例えば RAM 等の半導体メモリにより構成される記憶媒体であり、入力レート情報 691 や出力レート情報 692 等の情報を記憶する。

【0225】

[処理の流れ]

次に、以上のような復号装置 600 において実行される各種処理の流れについて説明する。最初に、制御部 601 による復号制御処理の流れの例を、図 26 のフローチャートを参照して説明する。

【0226】

復号制御処理が開始されると、制御部 601 の入力レート情報取得部 681 は、ステップ S601 において、入力レート情報を取得する。ステップ S602 において、出力レート情報取得部 682 は、出力レート情報を取得する。ステップ S603 において、復号処理制御部 683 は、コードストリームを入力系 611 (例えばキャッシュメモリ 643) から読み出し、画像復号部 612 に入力させる。

【0227】

ステップ S604 において、復号処理制御部 683 は、画像復号部 612 を制御し、入力させた画像データを復号する画像復号処理を実行させる。ステップ S605 において、負荷情報取得部 684 は、その復号処理についてスループット情報 693 を取得する。

【0228】

ステップ S606 において、負荷制御部 685 は、入力レートまたは出力レートのうち、少なくともいずれか一方に基づいて、ステップ S604 の制御によって実行された画像復号処理のスループットを制御する。

【0229】

ステップ S607 において、復号処理制御部 683 は、画像復号処理により生成された画像データを出力系 613 に出力し、処理させる。

【0230】

ステップ S608 において、復号処理制御部 683 は、復号制御処理を終了するか否かを判定する。例えば未処理の画像データが入力系 611 (例えばキャッシュメモリ 643) に存在する等して、復号制御処理を終了しないと判定された場合、ステップ S601 に処理が戻り、それ以降の処理が繰り返される。

【0231】

このとき、入力レートが固定値の場合、ステップ S601 の処理は省略可能である。同様に、出力レートが固定値の場合、ステップ S602 の処理は省略可能である。

【0232】

ステップ S608 において、例えば未処理の画像データが入力系 611 (例えばキャッシュメモリ 643) に存在せず、復号制御処理を終了すると判定された場合、復号制御処理が終了される。

【0233】

なお、ステップ S605 の処理 (スループット情報の取得) は、その情報の内容に応じて適切なタイミングで行われる。したがって、例えば、ステップ S604 の処理より先にステップ S605 およびステップ S606 の処理が行われるようにしてもよい。また、例えば、画像復号処理実行途中において、ステップ S605 およびステップ S606 の処理が行われるようにしてもよい。

【0234】

次に、図 26 のステップ S606 において実行される負荷制御処理の詳細な流れの例について、図 27 のフローチャートを参照して説明する。

【0235】

負荷制御処理が開始されると、負荷制御部 685 は、ステップ S621 において、入力

10

20

30

40

50

レート情報 6 9 1 およびスループット情報 6 9 3 に基づいて、復号処理のスループットが
入力レートより低いかなかを判定する。

【 0 2 3 6 】

復号処理のスループットが入力レートより低いと判定された場合、処理はステップ S 6
2 2 に進む。この場合、画像復号処理の処理速度が遅すぎて、入力系 6 1 1 における処理
がオーバフローする恐れがある。つまり、画像復号処理のハードウェア資源の消費量が少
なすぎる。ハードウェア資源に余裕があるのであれば、さらに画像復号処理がハードウェ
ア資源を消費するようにすることにより、入力系 6 1 1 におけるオーバフローの発生を抑
制し、復号装置 6 0 0 のスループットを向上させることができる。そこで、負荷制御部 6
8 5 は、ステップ S 6 2 2 において、画像復号部 6 1 2 の画像復号処理の処理速度を加速
させ、画像復号処理のスループットを入力レートまで上昇させる。ステップ S 6 2 2 の処
理が終了すると、図 2 6 のステップ S 6 0 6 に戻り、それ以降の処理が実行される。

10

【 0 2 3 7 】

復号処理のスループットが入力レートより低くないと判定された場合、処理はステップ
S 6 2 3 に進む。負荷制御部 6 8 5 は、ステップ S 6 2 3 において、入力レート情報 6 9
1 およびスループット情報 6 9 3 に基づいて、復号処理のスループットが入力レートより
高いかなかを判定する。

【 0 2 3 8 】

復号処理のスループットが入力レートより高いと判定された場合、処理はステップ S 6
2 4 に進む。この場合、画像復号処理の処理速度が速すぎて、入力系 6 1 1 からのコード
ストリームの供給が追いつかない。つまり、画像復号処理がハードウェア資源を不要に消
費している。そこで、負荷制御部 6 8 5 は、ステップ S 6 2 4 において、画像復号部 6 1
2 の画像復号処理の処理速度を減速させ、画像復号処理のスループットを入力レートまで
低下させる。ステップ S 6 2 4 の処理が終了すると、図 2 6 のステップ S 6 0 6 に戻り、
それ以降の処理が実行される。

20

【 0 2 3 9 】

復号処理のスループットが入力レートより高くないと判定された場合、処理はステップ
S 6 2 5 に進む。負荷制御部 6 8 5 は、ステップ S 6 2 5 において、出力レート情報 6 9
2 およびスループット情報 6 9 3 に基づいて、復号処理のスループットが出力レートより
高いかなかを判定する。

30

【 0 2 4 0 】

復号処理のスループットが出力レートより高いと判定された場合、処理はステップ S 6
2 6 に進む。この場合、画像復号処理の処理速度が速すぎて、出力系 6 1 3 における処理
がオーバフローする恐れがある。つまり、画像復号処理がハードウェア資源を不要に消費
している。そこで、負荷制御部 6 8 5 は、ステップ S 6 2 6 において、画像復号部 6 1 2
の画像復号処理の処理速度を減速させ、画像復号処理のスループットを出力レートまで低
下させる。ステップ S 6 2 6 の処理が終了すると、図 2 6 のステップ S 6 0 6 に戻り、そ
れ以降の処理が実行される。

【 0 2 4 1 】

復号処理のスループットが出力レートより高くないと判定された場合、負荷制御処理が
終了され、図 2 6 のステップ S 6 0 6 に戻り、それ以降の処理が実行される。

40

【 0 2 4 2 】

以上のような制御処理が行われることにより、復号装置 6 0 0 は、復号処理においてハ
ードウェア資源をより有効に活用することができる。

【 0 2 4 3 】

なお、図 2 7 においては、ステップ S 6 2 1、ステップ S 6 2 3、またはステップ S 6
2 5 の各判定処理を、図 2 7 に示されるフローチャートの順に行うように説明したが、各
判定条件の優先順位は任意であり、どの判定から行うようにしてもよい。また、複数の判
定を満たすときのみ復号処理のスループットの制御（例えば、ステップ S 6 2 2、ステッ
プ S 6 2 4、またはステップ S 6 2 6）を行うようにしてもよい。

50

【 0 2 4 4 】

< 6 . 第 6 の 実 施 の 形 態 >

[画 像 復 号 部 の 構 成]

上述したように、符号化方式は任意である。つまり、画像復号部 6 1 2 が行う画像復合処理の復号方法は任意である。以下に符号化方式がJPEG2000の場合について説明する。図 2 8 は、画像復号部 6 1 2 の主な構成例を示すブロック図である。図 2 8 に示されるように、画像復号部 6 1 2 は、パケット解読部 7 0 1、算術復号部 7 0 2、ビットモデリング部 7 0 3、ビットプレーン合成部 7 0 4、コードブロック合成部 7 0 5、ウェーブレット逆変換部 7 0 6、およびDCレベル逆シフト部 7 0 7を有する。

【 0 2 4 5 】

パケット解読部 7 0 1 は、矢印 7 3 1 に示されるように、符号化装置 1 0 0 より供給されるパケットを解読し、矢印 7 3 2 に示されるように、コードストリームを算術復号部 7 0 2 に供給する。また、パケット解読部 7 0 1 は、コードストリームのパケットを解読し、コードストリームのファイルサイズを抽出し、矢印 7 2 1 に示されるように、負荷情報として負荷情報取得部 6 8 4 に供給する。

【 0 2 4 6 】

算術復号部 7 0 2 およびビットモデリング部 7 0 3 は、EBCOT部 7 1 1 として動作し、入力されるコードストリームに対して、JPEG2000規格で定められたEBCOTと呼ばれるエントロピ復号を行う。

【 0 2 4 7 】

算術復号部 7 0 2 は、算術符号化部 2 0 7 に対応する方法でコードストリームを復号し、矢印 7 3 3 に示されるように、コンテキストをビットモデリング部 7 0 3 に供給する。ビットモデリング部 7 0 3 は、ビットモデリング部 2 0 6 に対応する方法で、ビットプレーンに展開されたウェーブレット係数を生成する。ビットモデリング部 7 0 3 は、生成したビットプレーン毎の係数データを、矢印 7 3 4 に示されるように、ビットプレーン合成部 7 0 4 に供給する。

【 0 2 4 8 】

ビットプレーン合成部 7 0 4 は、ビットプレーンに展開されたウェーブレット係数を合成する。ビットプレーン合成部 7 0 4 は、ビットプレーンを合成したウェーブレット係数を、矢印 7 3 5 に示されるように、コードブロック合成部 7 0 5 に供給する。

【 0 2 4 9 】

コードブロック合成部 7 0 5 は、供給されたビットプレーンを用いてコードブロック単位の係数データを生成し、さらにそれらを合成し、サブバンド毎の係数データを生成する。コードブロック合成部 7 0 5 は、矢印 7 3 6 に示されるように、それをウェーブレット逆変換部 7 0 6 に供給する。

【 0 2 5 0 】

ウェーブレット逆変換部 7 0 6 は、供給されたウェーブレット係数をウェーブレット逆変換し、ベースバンドの画像データを生成する。ウェーブレット逆変換部 7 0 6 は、生成したベースバンドの画像データを、矢印 7 3 7 に示されるように、DCレベル逆シフト部 7 0 7 に供給する。

【 0 2 5 1 】

DCレベル逆シフト部 7 0 7 は、その画像データのDC成分に対して、DCレベルシフト部 2 0 1 においてシフトした分を元に戻すDCレベル逆シフト処理を必要に応じて行う。DCレベル逆シフト部 7 0 7 は、DCレベル逆シフト処理後の画像データを、矢印 7 3 8 に示されるように、画像復号部 6 1 2 の外部（出力系 6 1 3）に出力する。

【 0 2 5 2 】

[制 御 部 の 構 成]

次に、このような画像復号部 6 1 2 を制御する制御部について説明する。図 2 9 は、制御部の主な構成例を示すブロック図である。図 2 9 において、制御部 6 0 1 は、基本的に図 2 5 を参照して説明した場合と同様の構成を有する。すなわち、制御部 6 0 1 は、復号

10

20

30

40

50

処理制御部 6 8 3 乃至記憶部 6 8 6 を有する。

【 0 2 5 3 】

ただし、この場合、制御部 6 0 1 の負荷情報取得部 6 8 4 は、負荷情報として、画像復号部 6 1 2 のパケット解読部 7 0 1 からコードストリームのファイルサイズ 7 5 1 を取得する。復号処理の難易度は、復号処理対象のコードストリームのファイルサイズの大きさにおおよそ比例する。つまり、ファイルサイズが大きいほど、復号処理の難易度は上がり、復号処理の負荷量は増大することが予想される。

【 0 2 5 4 】

記憶部 6 8 6 には、入力レート情報 6 9 1 や出力レート情報 6 9 2 の代わりに、テーブル情報 7 5 2 が記憶されている。テーブル情報 7 5 2 は、ファイルサイズ 7 5 1 から画像復号処理の処理速度の制御量を決定するためのテーブル情報である。このテーブル情報 7 5 2 において、コードストリームのファイルサイズの大きさと画像復号処理の処理速度の制御量との対応関係は、入力レートと出力レートを考慮して作成されている。

10

【 0 2 5 5 】

負荷制御部 6 8 5 は、記憶部 6 8 6 より読み出したテーブル情報 7 5 2 を用いて、負荷情報取得部 6 8 4 により取得されたファイルサイズ 7 5 1 の値に対応する画像復号処理の処理速度の制御量を求め、復号処理制御部 6 8 3 を制御する。このようにすることにより、負荷制御部 6 8 5 は、入力レート、出力レート、および負荷情報に基づいて、画像復号処理の負荷量を制御することになる。

【 0 2 5 6 】

したがって、上述した場合と同様に、復号装置 6 0 0 は、復号処理においてハードウェア資源をより有効に活用することができる。

20

【 0 2 5 7 】

なお、この場合、図 2 5 の入力レート情報取得部 6 8 1 および出力レート情報取得部 6 8 2 は、省略される。

【 0 2 5 8 】

[処理の流れ]

次にこの場合の復号制御処理の流れの例を図 3 0 のフローチャートを参照して説明する。

【 0 2 5 9 】

復号制御処理が開始されると、負荷情報取得部 6 8 4 は、ステップ S 7 0 1 において、パケット解読部 7 0 1 を制御し、処理対象のコードストリームのファイルサイズ 7 5 1 を取得する。ステップ S 7 0 2 において、負荷制御部 6 8 5 は、取得されたファイルサイズ 7 5 1 と、テーブル情報 7 5 2 に基づいて、スループットを制御する。

30

【 0 2 6 0 】

ステップ S 7 0 3 において、復号処理制御部 6 8 3 は、入力系 6 1 1 (例えばキャッシュメモリ 6 4 3) からコードストリームを画像復号部 6 1 2 に入力させる。

【 0 2 6 1 】

ステップ S 7 0 4 において、復号処理制御部 6 8 3 は、EBCOT部 7 1 1 等を制御し、負荷制御部 6 8 5 により決定されたスループットでエントロピ復号を行わせる。

40

【 0 2 6 2 】

ステップ S 7 0 5 において、復号処理制御部 6 8 3 は、ビットプレーン合成部 7 0 4 を制御し、復号処理により生成された係数データのビットプレーンを合成させる。ステップ S 7 0 6 において、復号処理制御部 6 8 3 は、コードブロック合成部 7 0 5 を制御し、係数データのコードブロックを合成させ、サブバンド毎の係数データを生成させる。

【 0 2 6 3 】

ステップ S 7 0 7 において、復号処理制御部 6 8 3 は、ウェーブレット逆変換部 7 0 6 を制御し、サブバンド毎の係数データをウェーブレット逆変換させ、ベースバンドの画像データを生成する。ステップ S 7 0 8 において、復号処理制御部 6 8 3 は、DCレベル逆シフト部 7 0 7 を制御し、画像データのDCレベルを逆シフトさせる。復号処理制御部 6 8 3

50

は、ステップ S 7 0 9 において、DCレベル逆シフト部 7 0 7 を制御し、処理済の画像データを出力系 6 1 3 に出力させる。

【 0 2 6 4 】

ステップ S 7 1 0 において、復号処理制御部 6 8 3 は、復号制御処理を終了するか否かを判定する。終了しないと判定された場合、ステップ S 7 0 1 に戻り、それ以降の処理が繰り返される。また、ステップ S 7 1 0 において、復号制御処理を終了すると判定された場合、復号制御処理が終了される。

【 0 2 6 5 】

このように制御が行われることにより、復号装置 6 0 0 は、復号処理においてハードウェア資源をより有効に活用することができる。また、復号装置 6 0 0 は、コードストリームのファイルサイズに基づいてこのような負荷制御を行うことができるので、容易に、ハードウェア資源をより有効に活用することができる。

【 0 2 6 6 】

なお、制御部 6 0 1 は、コードストリームのファイルサイズの代わりに、コードストリームの例えばメインヘッダ等に付加される負荷制御情報に基づいて、画像復号処理のスループット制御を行うようにしてもよい。この負荷制御情報は、上述したように、例えば、画像データの圧縮率、画像符号化処理時に割り当てられたハードウェア資源（コア数等）、または、スループットを制御したときのシステム環境情報（変更条件履歴情報）等により構成される。

【 0 2 6 7 】

いずれの場合においても、記憶部 6 8 6 に、負荷制御情報に対応するテーブル情報が予め用意されており、負荷制御部 6 8 5 は、そのテーブル情報を用いて、取得された負荷制御情報の値に対応する制御量を求める。

【 0 2 6 8 】

このようにすることにより、復号装置 6 0 0 は、容易に、ハードウェア資源をより有効に活用することができる。

【 0 2 6 9 】

以上における画像復号処理のスループットの制御方法は任意である。例えば、画像復号処理のクロックの速度を変更するようにしてもよいし、画像復号処理の実行に割り当てる時間を増減させても良いし、復号方法を変更するようにしてもよい。

【 0 2 7 0 】

また、例えば、画像復号部 6 1 2 の画像復号処理をソフトウェアにより実現させるようにしてもよく、その場合、制御部 6 0 1 が、画像復号処理に割り当てるハードウェア資源の量を増減させることにより、画像復号処理のスループットを制御するようにしてもよい。例えば、複数のCPU、複数のコアを有するCPU、またはマルチスレッド対応のCPUを用いて画像復号処理を実行させる場合、制御部 6 0 1 は、画像復号処理に割り当てるCPU数、コア数、またはスレッド数を増減させることにより、画像復号処理のスループットを制御するようにしてもよい。

【 0 2 7 1 】

< 7 . 第 7 の実施の形態 >

[制御部の構成]

以下に、複数コアを有するCPUを用いて画像復号処理を行う場合のハードウェア資源割り当て方法の様子について説明する。

【 0 2 7 2 】

図 3 1 は、図 1 5 に示されるような、複数コアのCPUを有するハードウェア資源を用いて画像復号処理を実行する場合の、制御部の主な構成例を示すブロック図である。

【 0 2 7 3 】

この場合の制御部 6 0 1 の構成も、基本的に図 2 5 を参照して説明した場合と同様であるが、負荷制御部 6 8 5 の代わりに、ハードウェア資源割り当て部 7 6 1 を有する。ハードウェア資源割り当て部 7 6 1 は、入力レート情報 6 9 1、出力レート情報 6 9 2、およ

10

20

30

40

50

びスループット情報 6 9 3 に基づいて、画像復号処理に割り当てるハードウェア資源、すなわち、サブCPUコア 4 4 2 のコア数を制御する。復号処理制御部 6 8 3 は、ハードウェア資源割り当て部 7 6 1 により割り当てられたサブCPUコア 4 4 2 を用いて画像復号処理を行う。

【 0 2 7 4 】

なお、ここでは、入力レート情報 6 9 1 および出力レート情報 6 9 2 が予め記憶部 6 8 6 に記憶されているものとし、入力レート情報取得部 6 8 1 および出力レート情報取得部 6 8 2 は省略されるものとする。もちろんこれらを有するにしてもよい。

【 0 2 7 5 】

[処理の流れ]

次に、この場合の復号制御処理の流れの例について、図 3 2 のフローチャートを参照して説明する。

【 0 2 7 6 】

復号制御処理が開始されると、制御部 6 0 1 の符号化処理制御部 6 8 3 は、ステップ S 7 3 1 において、コードストリームを入力系 6 1 1 (例えばキャッシュメモリ 6 4 3) から読み出し、画像復号部 6 1 2 に入力させる。

【 0 2 7 7 】

ステップ S 7 3 2 において、復号処理制御部 6 8 3 は、画像復号部 6 1 2 を制御し、入力させたコードストリームを復号する画像復号処理を実行させる。ステップ S 7 3 3 において、負荷情報取得部 6 8 4 は、その復号処理についてスループット情報 6 9 3 を取得する。

【 0 2 7 8 】

ステップ S 7 3 4 において、ハードウェア資源割り当て部 7 6 1 は、入力レートまたは出力レートのうち、少なくともいずれか一方に基づいて、画像復号処理の一部または全部にハードウェア資源を割り当てる。この時、例えば復号方法がJPEG2000の場合、復号処理全体に割り当てるハードウェア資源を制御するようにしても良いし、EBCOT部 7 1 1 1 により行われるエントロピ復号処理に割り当てるハードウェア資源を制御するようにしてもよい。

【 0 2 7 9 】

ステップ S 7 3 5 において、復号処理制御部 6 8 3 は、画像復号処理により生成された画像データを出力系 6 1 3 に出力し、処理させる。ステップ S 7 3 6 において、復号処理制御部 6 8 3 は、復号制御処理を終了するか否かを判定する。例えば未処理のコードストリームが入力系 6 1 1 (例えばキャッシュメモリ 6 4 3) に存在する等して、復号制御処理を終了しないと判定された場合、ステップ S 7 3 1 に処理が戻り、それ以降の処理が繰り返される。

【 0 2 8 0 】

ステップ S 7 3 6 において、例えば未処理のコードストリームが入力系 6 1 1 (例えばキャッシュメモリ 6 4 3) に存在せず、復号制御処理を終了すると判定された場合、復号制御処理が終了される。

【 0 2 8 1 】

なお、ステップ S 7 3 3 の処理(スループット情報の取得)は、その情報の内容に応じて適切なタイミングで行われる。したがって、例えば、ステップ S 7 3 2 の処理より先にステップ S 7 3 3 およびステップ S 7 3 4 の処理が行われるようにしてもよい。また、例えば、画像復号処理実行途中において、ステップ S 7 3 3 およびステップ S 7 3 4 の処理が行われるようにしてもよい。

【 0 2 8 2 】

以上のような制御処理により、制御部 6 0 1 は、例えば、画像復号処理のスループットが低すぎる場合、画像復号処理に割り当てるハードウェア資源(例えばコア数)を増やすことにより、スループットを向上させることができる。また、例えば、画像復号処理のスループットが高すぎる場合、制御部 6 0 1 は、画像復号処理に割り当てるハードウェア資

10

20

30

40

50

源（例えばコア数）を減らすことにより、スループットを低減させることができる。

【0283】

このように、制御部601は、入力レートや出力レート等に基づいて、画像復号処理に割り当てるハードウェア資源の量をより適切な状態に保つことができる。画像復号処理に割り当てられないコアは、例えば他の処理に割り当てることができるので、以上のような制御により、サブCPUコア442がより有効に活用されるようになる。つまり、復号装置600は、画像復号処理においてハードウェア資源をより有効に活用することができる。

【0284】

なお、画像復号処理に割り当てるCPUの数を制御する場合や、画像復号処理に割り当てるスレッド数を制御する場合も、基本的にコア数を制御する場合と同様に制御される。

10

【0285】

上述した一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、ネットワークや記録媒体からインストールされる。この記録媒体は、例えば、図15に示されるように、装置本体とは別に、ユーザにプログラムを配信するために配布される、プログラムが記録されている磁気ディスク（フレキシブルディスクを含む）、光ディスク（CD-ROM（Compact Disc - Read Only Memory）、DVD（Digital Versatile Disc）を含む）、光磁気ディスク（MD（Mini Disc）を含む）、もしくは半導体メモリなどよりなるリムーバブルメディア311により構成されるだけでなく、装置本体に予め組み込まれた状態でユーザに配信される、プログラムが記録されているHDD405や記憶装置303に含まれるハードディスクやメモリなどで構成される。

20

【0286】

なお、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【0287】

もちろん、上述した一連の処理は、汎用のハードウェアを用いてソフトウェアを実行することにより実現させることもできるが、専用のハードウェアにより実現させることもできる。

【0288】

また、本明細書において、システムとは、複数のデバイス（装置）により構成される装置全体を表すものである。

30

【0289】

なお、以上において、1つの装置（または処理部）として説明した構成を分割し、複数の装置（または処理部）として構成するようにしてもよい。逆に、以上において複数の装置（または処理部）として説明した構成をまとめて1つの装置（または処理部）として構成されるようにしてもよい。また、各装置（または処理部）の構成に上述した以外の構成を付加するようにしてももちろんよい。さらに、システム全体としての構成や動作が実質的に同じであれば、ある装置（または処理部）の構成の一部を他の装置（または処理部）の構成に含めるようにしてもよい。つまり、本発明の実施の形態は、上述した実施の形態に限定されるものではなく、本発明の要旨を逸脱しない範囲において種々の変更が可能である。

40

【符号の説明】

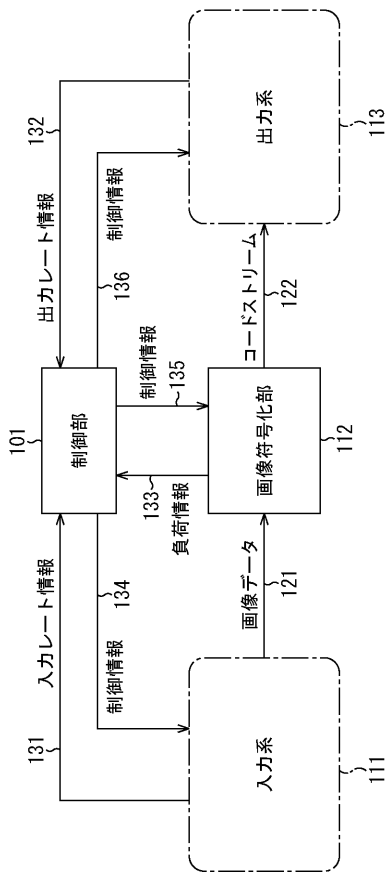
【0290】

100 符号化装置， 101 制御部， 111 入力系， 112 画像符号化部， 113 出力系， 183 符号化処理制御部， 184 負荷情報取得部， 185 負荷制御部， 191 スループット情報， 251 ゼロビットプレーン数， 252 テーブル情報， 461 ハードウェア資源割り当て部， 471 情報付加部， 600 復号装置， 601 制御部， 611 入力系， 612 画像復号部， 613 出力系， 683 復号処理制御部， 684 負荷情報取得部， 685 付加制御部， 693 スループット情報， 751 ファイルサイズ， 752 テーブ

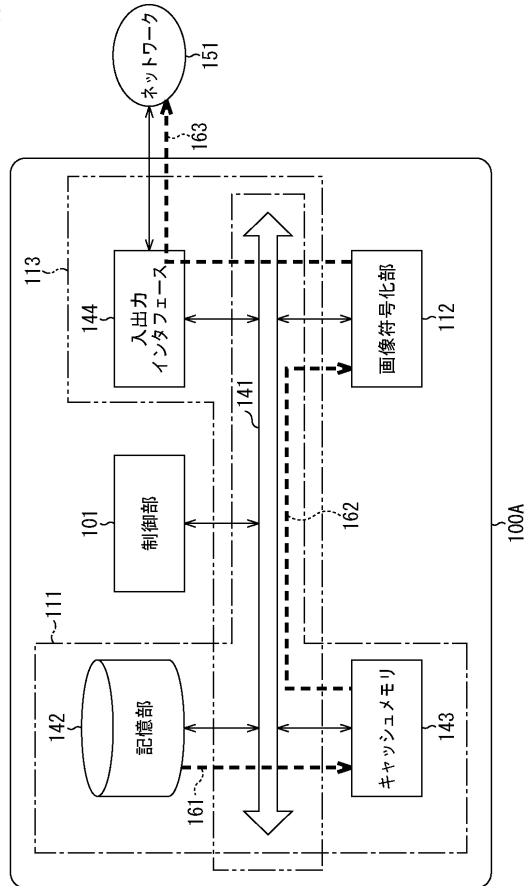
50

ル情報, 761 ハードウェア資源割り当て部

【図1】



【図2】

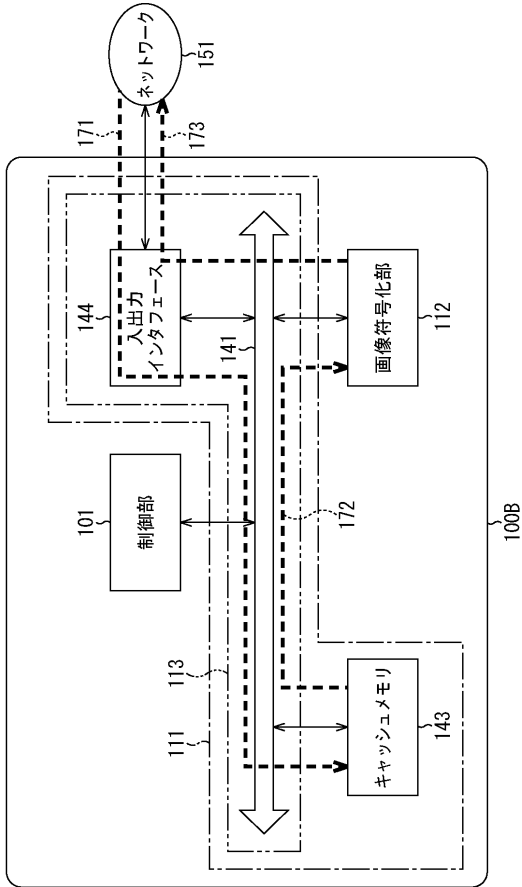


100 符号化装置

100A

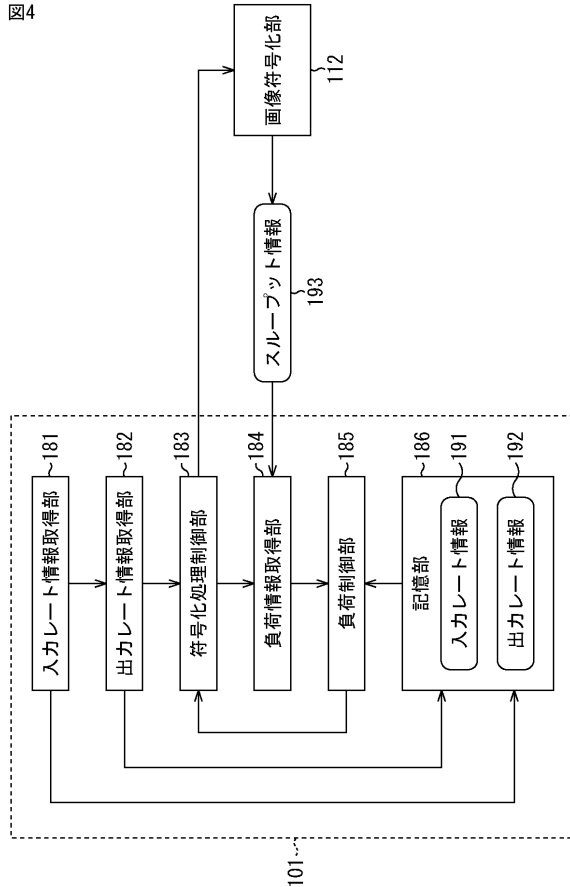
【図3】

図3



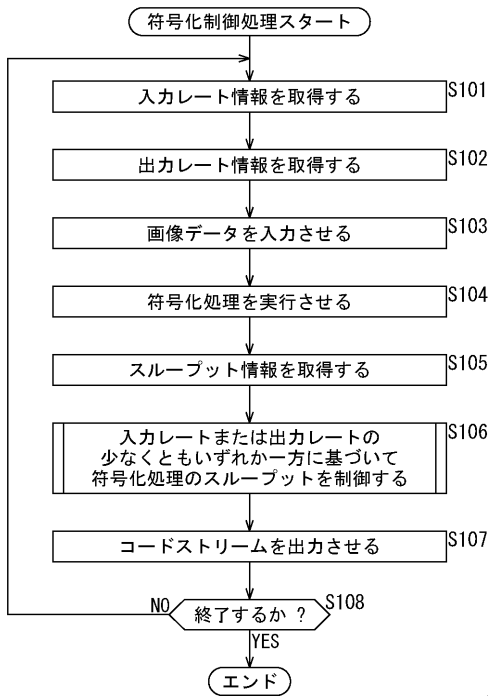
【図4】

図4



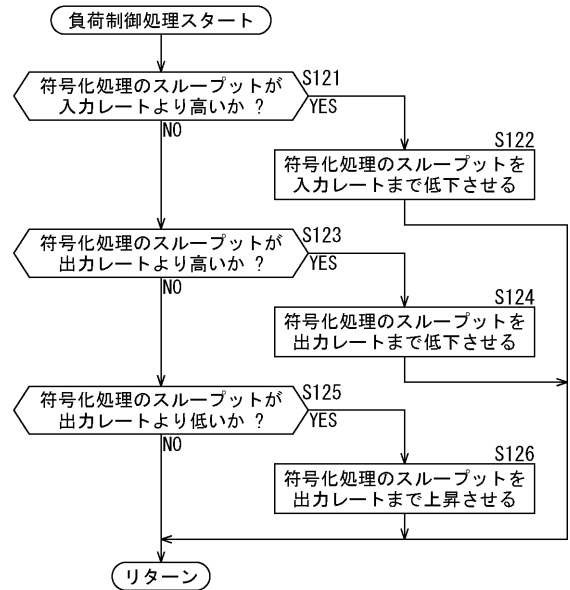
【図5】

図5

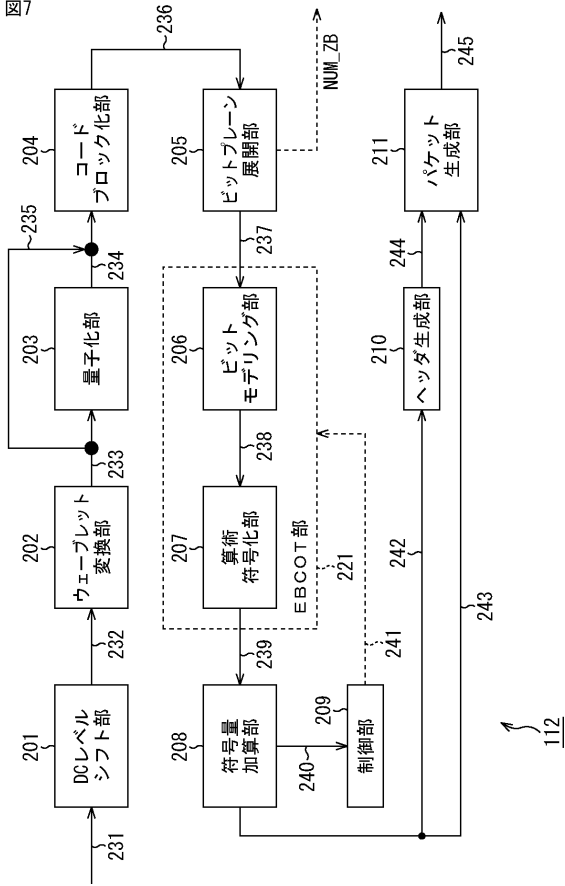


【図6】

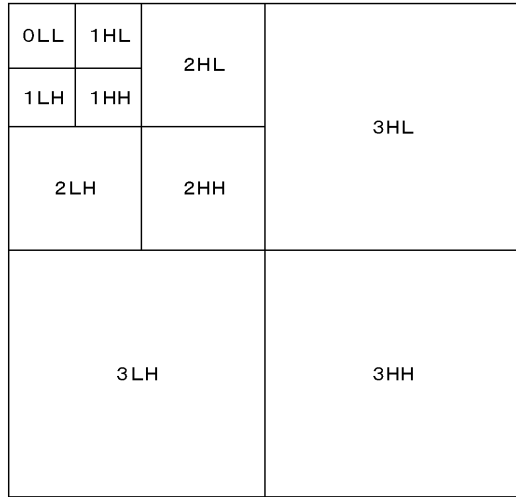
図6



【図7】

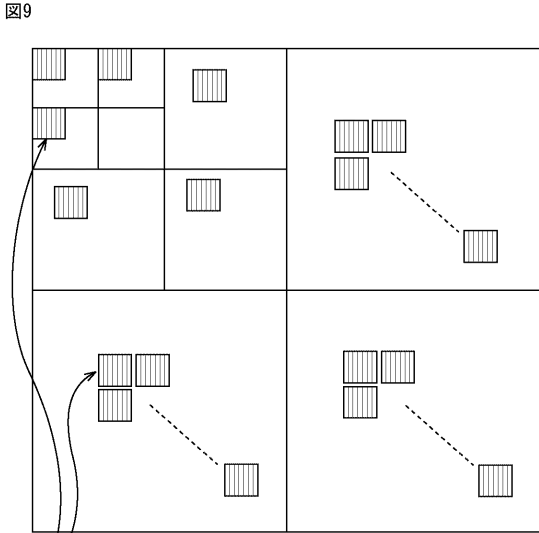


【図8】



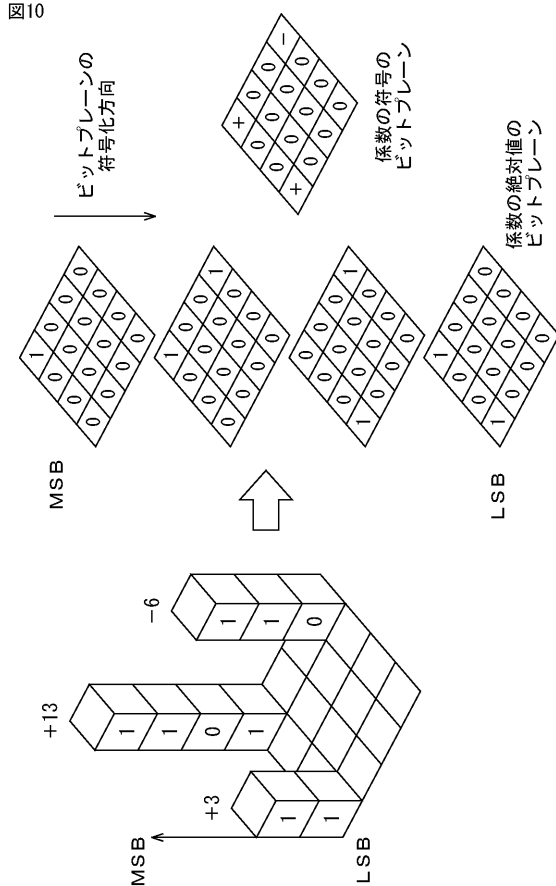
分割レベル=3
(H:高域、L:低域)

【図9】



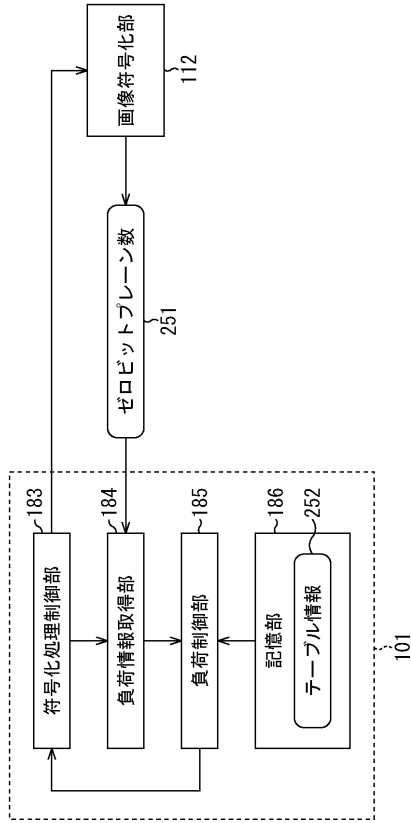
コードブロック
(例) 64×64

【図10】



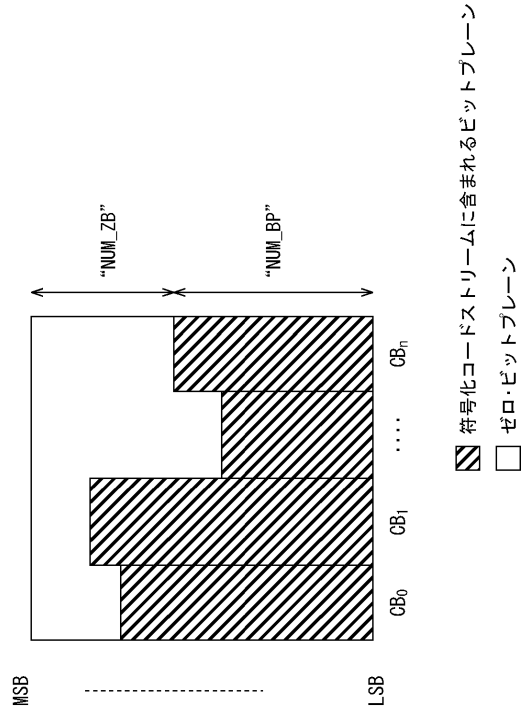
【図 1 1】

図11



【図 1 2】

図12



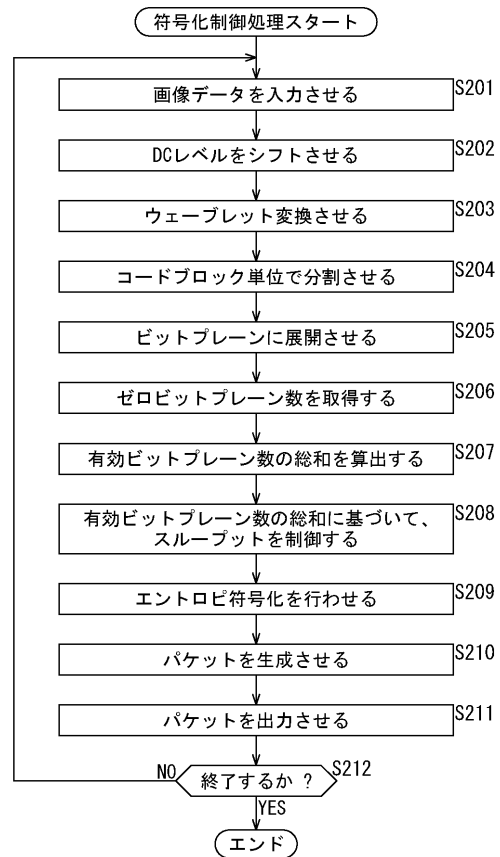
【図 1 3】

図13

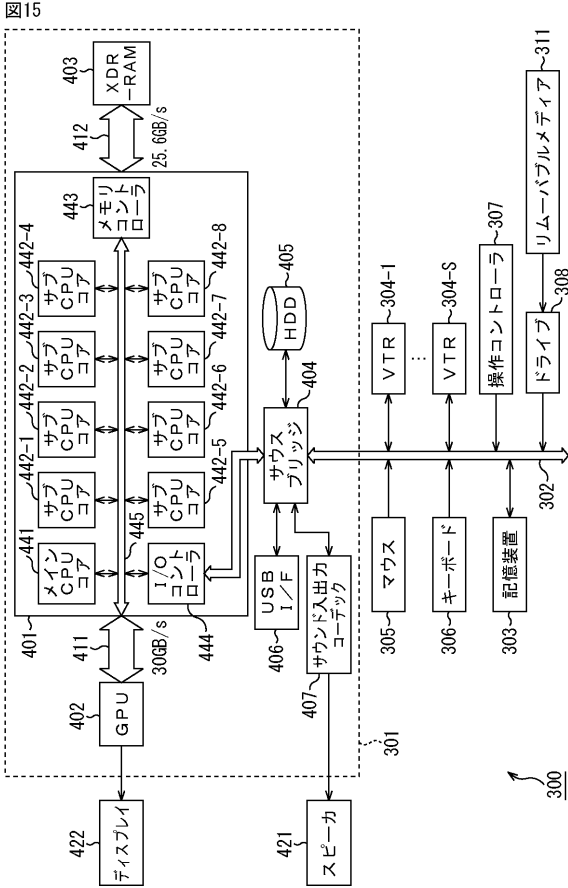
252	
"ALL_NUM_BP"	"Target_Speed"
~50	X0.02
51~100	X0.1
101~500	X0.5
501~1000 (= "Ref_ALL_NUM_BP")	X1.0
1001~1500	X1.5
1501~1800	X1.8
1801~	X2.0

【図 1 4】

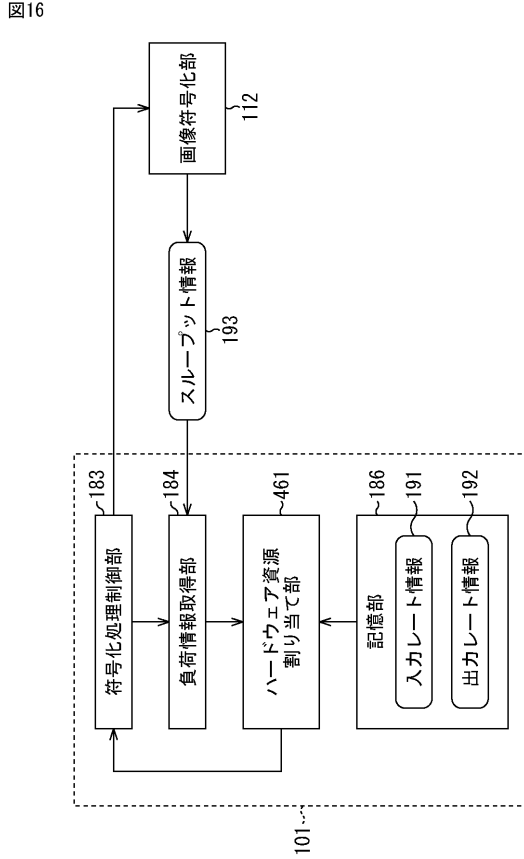
図14



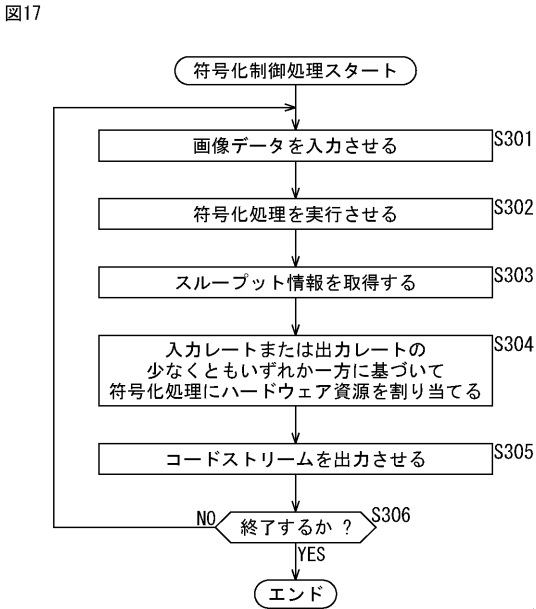
【図15】



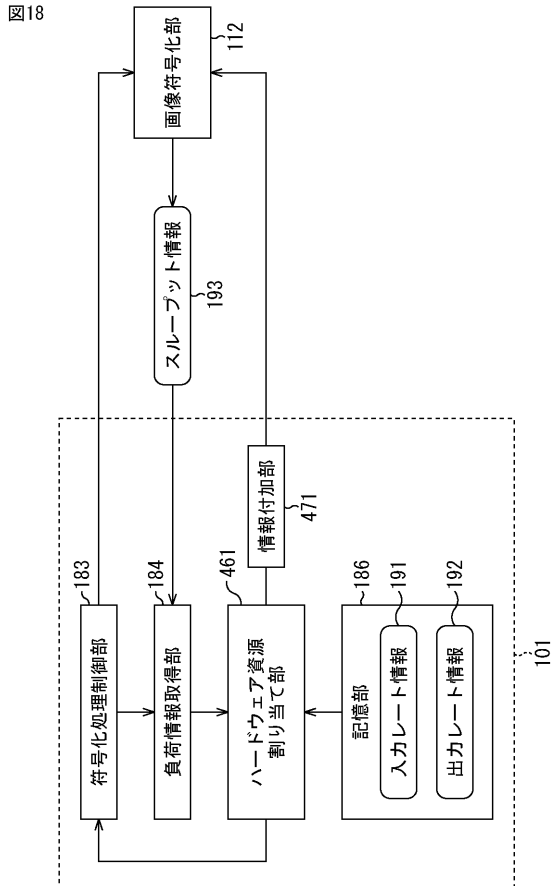
【図16】



【図17】

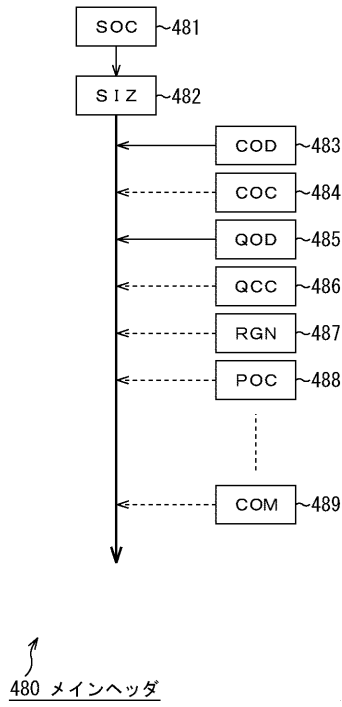


【図18】



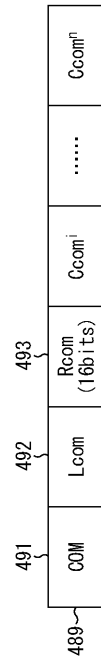
【図19】

図19



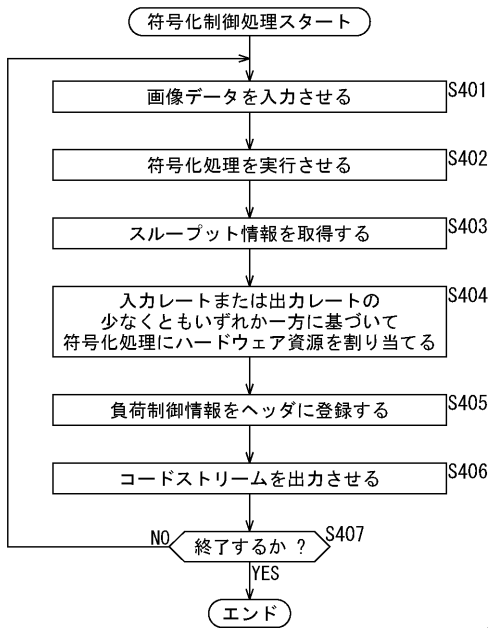
【図20】

図20



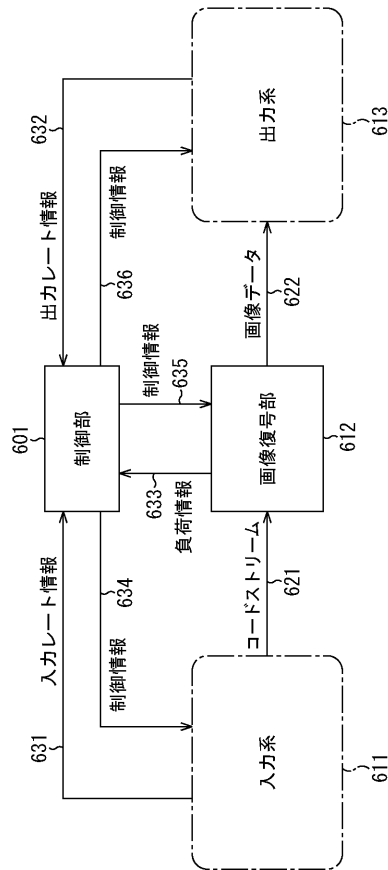
【図21】

図21



【図22】

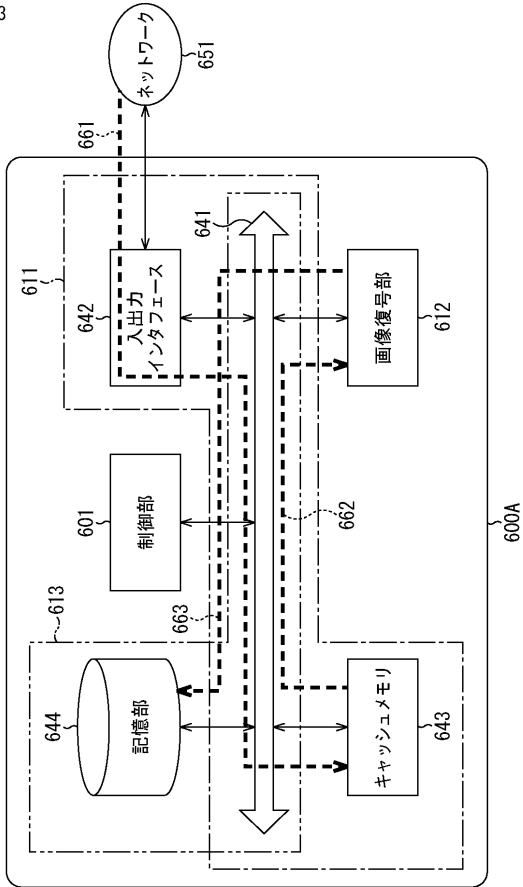
図22



600 復号装置

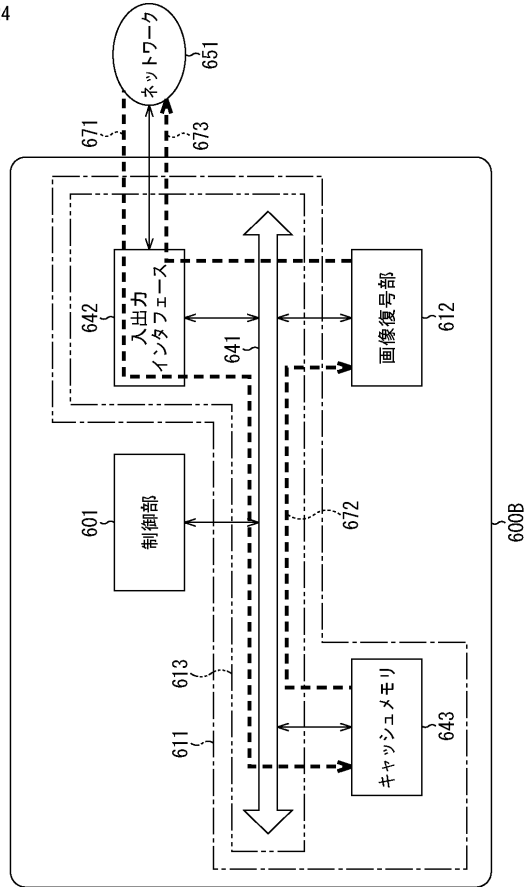
【図 2 3】

図23



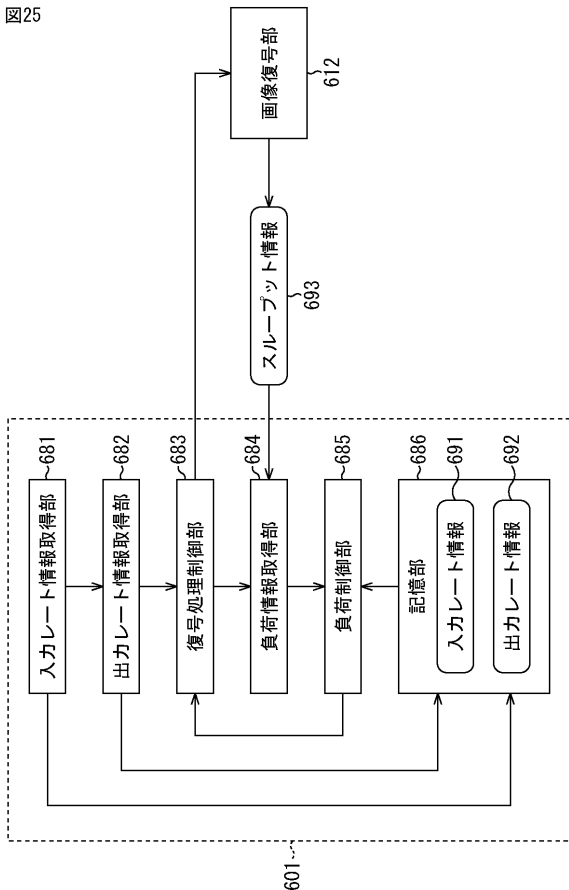
【図 2 4】

図24



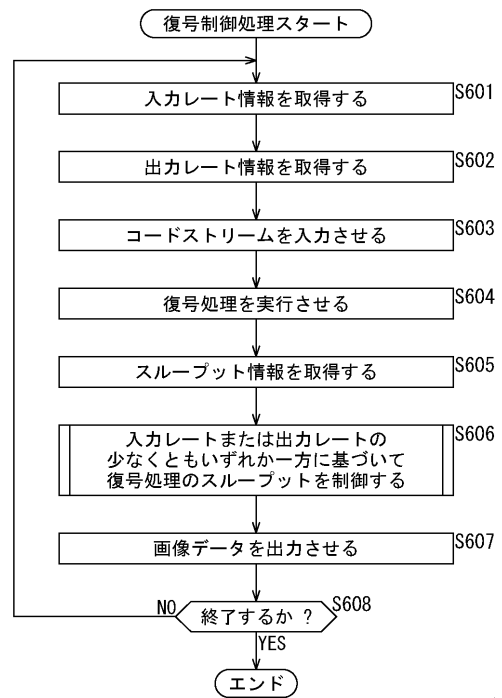
【図 2 5】

図25



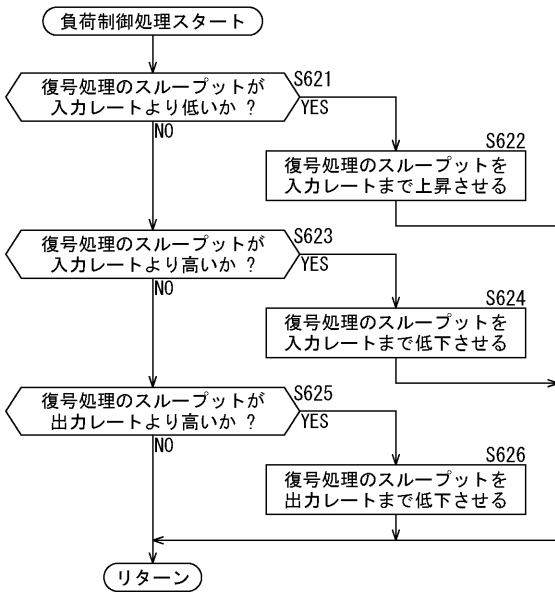
【図 2 6】

図26



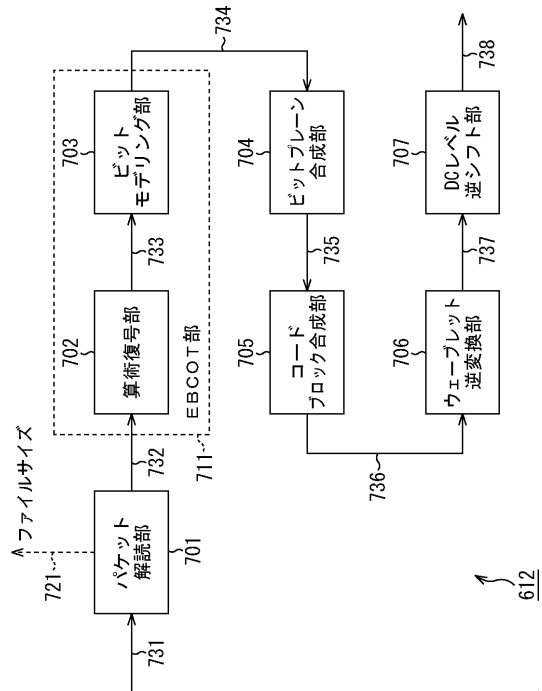
【図 27】

図27



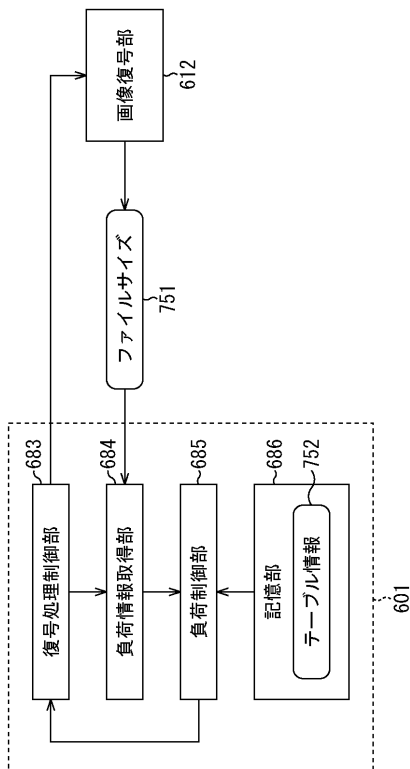
【図 28】

図28



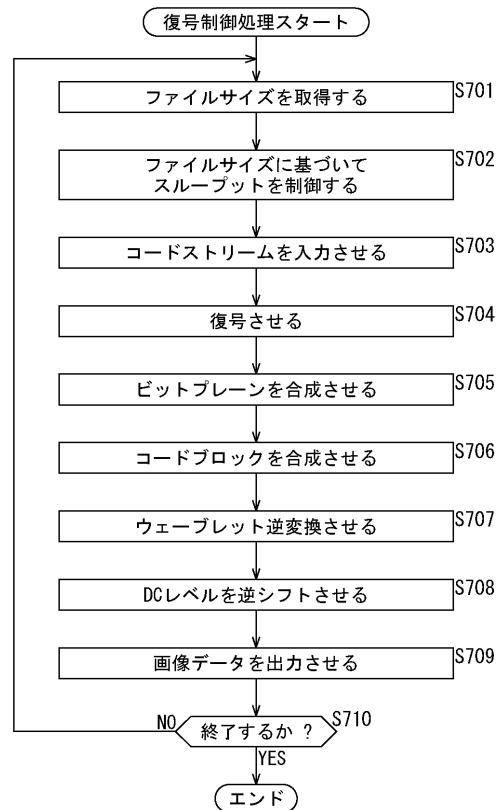
【図 29】

図29



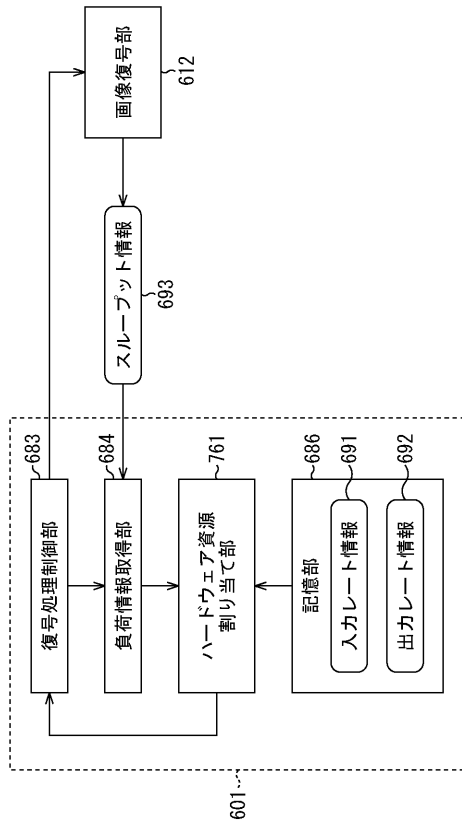
【図 30】

図30



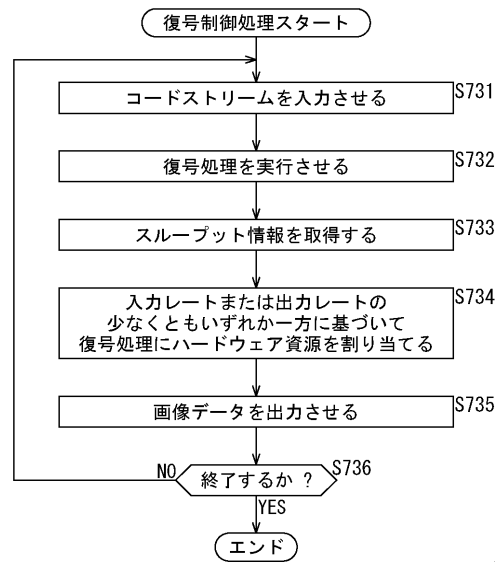
【図 3 1】

図31



【図 3 2】

図32



フロントページの続き

Fターム(参考) 5C159 KK13 MA00 MA35 MA41 MC11 MC38 ME11 PP01 PP04 SS08
TA00 TC00 TD11 UA02 UA05
5C178 AC07 BC01 BC62 BC93 DC47 DC65 EC57 FC02 GC02 HC03
5J064 AA03 AA04 BA09 BA16 BC01 BC14 BC16 BD02 BD03