

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7485705号
(P7485705)

(45)発行日 令和6年5月16日(2024.5.16)

(24)登録日 令和6年5月8日(2024.5.8)

(51)国際特許分類	F I
B 2 5 J 9/10 (2006.01)	B 2 5 J 9/10 A
G 0 5 B 19/42 (2006.01)	G 0 5 B 19/42 Z
B 2 5 J 9/22 (2006.01)	B 2 5 J 9/22 A
G 0 5 B 19/4069(2006.01)	G 0 5 B 19/4069

請求項の数 22 (全29頁)

(21)出願番号	特願2022-19414(P2022-19414)	(73)特許権者	000006622
(22)出願日	令和4年2月10日(2022.2.10)		株式会社安川電機
(65)公開番号	特開2023-116978(P2023-116978 A)		福岡県北九州市八幡西区黒崎城石 2 番 1 号
(43)公開日	令和5年8月23日(2023.8.23)	(74)代理人	100088155
審査請求日	令和4年12月21日(2022.12.21)		弁理士 長谷川 芳樹
		(74)代理人	100145012
			弁理士 石坂 泰紀
		(74)代理人	100171099
			弁理士 松尾 茂樹
		(72)発明者	前田 貴宏
			福岡県北九州市八幡西区黒崎城石 2 番 1 号 株式会社安川電機内
		(72)発明者	丸野 元春
			福岡県北九州市八幡西区黒崎城石 2 番 1 号

最終頁に続く

(54)【発明の名称】 ロボットシステム、タスク生成装置及び制御方法

(57)【特許請求の範囲】

【請求項 1】

未定区間を含むロボットの動作パスを表す複数のコマンドを順次呼び出す呼出部と、
前記未定区間に対する追加パスを生成するパスプラン部と、
前記呼出部が呼び出したコマンドと、前記追加パスとに基づいて前記ロボットを動作させる制御部と、
を備え、

前記パスプラン部は、1 のコマンドに基づいて前記制御部が前記ロボットを動作させている際に、前記ロボットの周辺環境情報に基づいて前記追加パスを生成する、ロボットシステム。

【請求項 2】

前記複数のコマンドは、前記動作パスの経由点の情報を含むムーブコマンドと、前記未定区間の到達点となる前記動作パスの経由点の情報を含むオートコマンドと、を含み、

前記パスプラン部は、前記ムーブコマンドに基づき前記制御部が前記ロボットを動作させている際に、前記ムーブコマンドより後の前記オートコマンドと前記周辺環境情報とに基づいて、前記未定区間における前記到達点までの前記追加パスを生成する、請求項 1 記載のロボットシステム。

【請求項 3】

2 以上のムーブコマンドをそれぞれ含む複数のタスクを保存するタスク保存部と、
前記タスク保存部に保存された前記複数のタスクのうち、1 のタスクを選択するタスク

選択部と、
を更に備え、

前記呼出部は、前記タスク選択部が選択した前記 1 のタスクに含まれる前記ムーブコマンドを呼び出す前に、前記オートコマンドを呼び出す、請求項 2 記載のロボットシステム。

【請求項 4】

前記複数のタスクの少なくともいずれかは、前記 2 以上のムーブコマンドの前に前記オートコマンドを含み、

前記呼出部は、前記タスク選択部が選択した前記 1 のタスクから前記オートコマンドを呼び出す、請求項 3 記載のロボットシステム。

【請求項 5】

前記 2 以上のムーブコマンドを指定するタスク情報に基づいて、前記 2 以上のムーブコマンドの前に実行される前記オートコマンドを生成するコマンド生成部と、

生成された前記オートコマンドと、前記タスク情報とに基づいてタスクを生成するタスク生成部と、

を更に備える、請求項 4 記載のロボットシステム。

【請求項 6】

前記 1 のタスクにおいて最初の経由点が前記ムーブコマンドにより定められている場合に、前記最初の経由点を定める前記ムーブコマンドを前記オートコマンドに置き換えるか、前記最初の経由点を定める前記ムーブコマンドの前に前記オートコマンドを挿入する、オートコマンド配置部を更に有する、請求項 4 記載のロボットシステム。

【請求項 7】

前記呼出部は、前記 1 のタスクにおいて最初の経由点が前記ムーブコマンドにより定められている場合に、前記最初の経由点を定める前記ムーブコマンドを、前記最初の経由点を前記到達点とする前記オートコマンドに読み替える、請求項 3 記載のロボットシステム。

【請求項 8】

前記パスプラン部は、前記到達点の直前の前記経由点から前記到達点までの前記追加パスを生成する、請求項 2 ～ 7 のいずれか一項記載のロボットシステム。

【請求項 9】

前記複数のコマンドは、前記経由点をシフトさせるシフトコマンドを含み、

前記パスプラン部は、前記呼出部が、前記シフトコマンドの後に前記オートコマンドを呼び出した場合に、前記オートコマンドの前記到達点を前記シフトコマンドに基づきシフトさせたシフト到達点までの前記追加パスを生成する、請求項 8 記載のロボットシステム。

【請求項 10】

前記パスプラン部は、前記オートコマンドの直前の前記ムーブコマンドに対応する動作が完了する前に前記追加パスの生成が完了するタイミングで、前記追加パスの生成を開始する、請求項 2 ～ 9 のいずれか一項記載のロボットシステム。

【請求項 11】

前記パスプラン部は、前記周辺環境情報に基づいて、前記追加パスを定める 2 以上の新たなムーブコマンドを生成する、請求項 2 ～ 10 のいずれか一項記載のロボットシステム。

【請求項 12】

前記パスプラン部は、前記呼出部が前記オートコマンドを呼び出した後に、前記オートコマンドと前記周辺環境情報とに基づいて、前記追加パスを定める前記 2 以上の新たなムーブコマンドを生成し、

前記呼出部は、前記 2 以上の新たなムーブコマンドを含む 2 以上のコマンドをコマンド保存部に記憶させ、

前記制御部は、前記コマンド保存部が記憶する前記 2 以上のコマンドに基づいて前記ロボットを動作させる、請求項 11 記載のロボットシステム。

【請求項 13】

前記制御部は、

前記コマンド保存部が記憶する前記 2 以上のコマンドに基づいて前記ロボットの一連

10

20

30

40

50

の制御指令を生成し、

一連の制御指令に基づいて前記ロボットを動作させる、請求項 1 2 記載のロボットシステム。

【請求項 1 4】

前記呼出部は、前記コマンド保存部が記憶する前記 2 以上のコマンドに対応する動作の予想時間が、少なくとも前記パスプラン部による前記追加パスの生成に要する時間よりも長くなるように、前記コマンド保存部に記憶させる前記コマンドの数を変更する、請求項 1 2 又は 1 3 記載のロボットシステム。

【請求項 1 5】

前記オートコマンドは、前記周辺環境情報に基づいて前記追加パスを生成する際の生成条件を表す条件情報を含み、

前記パスプラン部は、前記条件情報に更に基づいて前記追加パスを生成する、請求項 2 ~ 1 4 のいずれか一項記載のロボットシステム。

【請求項 1 6】

前記パスプラン部が生成した前記追加パスに基づく前記ロボットの動作をシミュレートして、前記ロボットが前記周辺環境に存在する周辺物体に干渉しないか確認する干渉チェック部を更に有し、

前記制御部は、前記ロボットが前記周辺物体に干渉しない場合に、前記パスプラン部が生成した前記追加パスに基づいて前記ロボットを動作させる、請求項 1 ~ 1 5 のいずれか一項記載のロボットシステム。

【請求項 1 7】

少なくとも、前記パスプラン部が前記追加パスを生成する前後に、前記周辺環境情報を更新する情報収集部と、

前記周辺環境情報に基づいて、前記パスプラン部が前記追加パスを生成する前後における前記周辺環境情報の変化がないかを確認する環境変化チェック部と、
を更に備える、請求項 1 ~ 1 6 のいずれか一項記載のロボットシステム。

【請求項 1 8】

前記制御部は、前記パスプラン部が前記追加パスを生成する前後における前記周辺環境情報の変化がない場合に、前記追加パスに基づいて前記ロボットを動作させる、請求項 1 7 記載のロボットシステム。

【請求項 1 9】

前記パスプラン部は、前記追加パスを生成する前後における前記周辺環境情報の変化がある場合に、変化後の前記周辺環境情報に基づいて前記追加パスを再生成する、請求項 1 7 又は 1 8 記載のロボットシステム。

【請求項 2 0】

未定区間を含むロボットの動作パスの経由点の情報を含む複数のムーブコマンドと、前記未定区間の到達点となる前記動作パスの経由点の情報を含むオートコマンドと、を含む複数のコマンドを順次呼び出す呼出部と、

前記呼出部が前記オートコマンドを呼び出した場合に、前記オートコマンドと、前記ロボットの周辺環境情報とに基づいて、前記未定区間における前記到達点までの追加パスを生成するパスプラン部と、

前記呼出部が呼び出したコマンドと、前記追加パスとに基づいて前記ロボットを動作させる制御部と、
を備えるロボットシステム。

【請求項 2 1】

ロボットの動作パスの経由点の情報をそれぞれが含む 2 以上のムーブコマンドを指定するタスク情報に基づいて、前記 2 以上のムーブコマンドに対応する動作への移行動作における到達点の情報を含むオートコマンドを生成するコマンド生成部と、

生成した前記オートコマンドと、前記タスク情報とに基づいてタスクを生成するタスク生成部と、を備え、

10

20

30

40

50

前記タスクを前記ロボットに実行させるために前記オートコマンドが呼び出されると、前記到達点までの追加パスが前記ロボットの周辺環境情報に基づいて生成され、生成された前記追加パスと前記タスクの前記 2 以上のムーブコマンドとに基づいて前記ロボットが動作する、タスク生成装置。

【請求項 22】

未定区間を含むロボットの動作パスを表す複数のコマンドを順次呼び出すことと、
前記未定区間に対する追加パスを生成することと、
呼び出したコマンドと、前記追加パスとに基づいて前記ロボットを動作させることと、
を含み、

1 のコマンドに基づいて前記ロボットを動作させている際に、周辺環境情報に基づいて前記追加パスを生成する、制御方法。

10

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、ロボットシステム、タスク生成装置及び制御方法に関する。

【背景技術】

【0002】

特許文献 1 には、ロボットと作業環境の幾何学的形状およびそれらの配置を記述した計算機上の幾何モデル手段と、モデル同士の干渉を検査する計算機上の干渉検査手段を利用し、ロボットのスタートおよびゴール配置が与えられたとき、ロボットと作業環境内の障害物とが干渉しないロボットの動作経路を計画する方法が開示されている。

20

【先行技術文献】

【特許文献】

【0003】

【文献】特開 2000 - 20117 号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

本開示は、動作プログラミングの簡素化に有効なロボットシステムを提供する。

【課題を解決するための手段】

30

【0005】

本開示の一側面に係るロボットシステムは、未定区間を含むロボットの動作パスを表す複数のコマンドを順次呼び出す呼出部と、未定区間に対する追加パスを生成するパスプラン部と、呼出部が呼び出したコマンドと、追加パスとに基づいてロボットを動作させる制御部と、を備え、パスプラン部は、1 のコマンドに基づいて制御部がロボットを動作させている際に、ロボットの周辺環境情報に基づいて追加パスを生成する。

【0006】

本開示の他の一側面に係るロボットシステムは、未定区間を含むロボットの動作パスの経由点の情報を含む複数のムーブコマンドと、前記未定区間の到達点となる前記動作パスの経由点の情報を含むオートコマンドと、を含む複数のコマンドを順次呼び出す呼出部と、オートコマンドと、ロボットの周辺環境情報とに基づいて、未定区間における到達点までの追加パスを生成するパスプラン部と、呼出部が呼び出したコマンドと、追加パスとに基づいてロボットを動作させる制御部と、を備える。

40

【0007】

本開示の更に他の一側面に係るタスク生成装置は、未定区間を含むロボットの動作パスの経由点の情報をそれぞれが含む 2 以上のムーブコマンドを指定するタスク情報に基づいて、2 以上のムーブコマンドに対応する一連の動作の開始点を定めるオートコマンドを生成するコマンド生成部と、生成したオートコマンドと、タスク情報とに基づいてタスクを生成するタスク生成部と、を備え、タスクが選択されると、タスクのオートコマンドが定める開始点までの追加パスがロボットの周辺環境情報に基づいて生成され、生成された追

50

加パスとタスクの２以上のムーブコマンドとに基づいてロボットが動作する。

【０００８】

本開示の更に他の一側面に係る制御方法は、未定区間を含むロボットの動作パスを表す複数のコマンドを順次呼び出すことと、未定区間に対する追加パスを生成することと、呼び出したコマンドと、追加パスとに基づいてロボットを動作させることと、を含み、１のコマンドに基づいてロボットを動作させている際に、周辺環境情報に基づいて追加パスを生成する。

【発明の効果】

【０００９】

本開示によれば、動作プログラミングの簡素化に有効なロボットシステムを提供することができる。

10

【図面の簡単な説明】

【００１０】

【図１】ロボットシステムの構成を例示する模式図である。

【図２】ロボットの構成を例示する模式図である。

【図３】コントローラの機能的な構成を例示するブロック図である。

【図４】複数のコマンドを例示する図である。

【図５】パスプラン部の構成を例示するブロック図である。

【図６】ロボットの動作期間と、追加パスの生成期間との関係を表すタイミングチャートである。

20

【図７】タスク保存部の記憶内容を例示する模式図である。

【図８】コントローラの変形例を示すブロック図である。

【図９】上位コントローラの構成を例示するブロック図である。

【図１０】コントローラ、上位コントローラ、及びシミュレーション装置のハードウェア構成を例示するブロック図である。

【図１１】オートコマンド配置手順を例示するフローチャートである。

【図１２】タスク生成手順を例示するフローチャートである。

【図１３】プログラムの生成手順を例示するフローチャートである。

【図１４】システム制御手順を例示するフローチャートである。

【図１５】コマンド呼出手順を例示するフローチャートである。

30

【図１６】パス生成手順を例示するフローチャートである。

【図１７】環境変化チェック手順を例示するフローチャートである。

【図１８】コマンドに基づくロボットの制御手順を例示するフローチャートである。

【発明を実施するための形態】

【００１１】

以下、実施形態について、図面を参照しつつ詳細に説明する。説明において、同一要素又は同一機能を有する要素には同一の符号を付し、重複する説明を省略する。

【００１２】

〔ロボットシステム〕

図１に示すロボットシステム１は、１以上のロボットを含む複数のマシンによりワークの生産等を行うシステムである。ロボットシステム１が複数のマシンに実行させる内容は、ワーク等の生産に限られない。一例として、ロボットシステム１は、コンベヤ２と、複数のロボット３と、制御システムＣＳ１とを備える。

40

【００１３】

コンベヤ２は、例えば電動モータ等の動力によりワークを搬送する。コンベヤ２の具体例としては、ベルトコンベヤ、ローラコンベヤ等が挙げられる。複数のロボット３のそれぞれは、コンベヤ２が搬送するワークに対する作業を行う。ワークに対する作業の具体例としては、コンベヤ２が搬送するワーク（例えばベースパーツ）に対する他のワーク（例えばサブパーツ）の組付け、コンベヤ２が搬送するワークにおけるパーツ同士の締結（例えばボルト締結）・接合（例えば溶接）等が挙げられる。

50

【 0 0 1 4 】

図 2 は、ロボット 3 の構成を例示する模式図である。図 2 に示すロボット 3 は、6 軸の垂直多関節ロボットであり、基部 1 1 と、旋回部 1 2 と、第 1 アーム 1 3 と、第 2 アーム 1 4 と、第 3 アーム 1 7 と、先端部 1 8 と、アクチュエータ 4 1 , 4 2 , 4 3 , 4 4 , 4 5 , 4 6 とを有する。基部 1 1 は、コンベヤ 2 の周囲に設置されている。旋回部 1 2 は、鉛直な軸線 2 1 まわりに回転するように基部 1 1 上に設けられている。第 1 アーム 1 3 は、軸線 2 1 に交差（例えば直交）する軸線 2 2 まわりに揺動するように旋回部 1 2 に接続されており、軸線 2 2 から離れる方向に向かって延びている。交差は、所謂立体交差のようにねじれの関係にある場合も含む。以下においても同様である。

【 0 0 1 5 】

第 2 アーム 1 4 は、軸線 2 2 に実質的に平行な軸線 2 3 まわりに揺動するように第 1 アーム 1 3 の先端部に接続されており、軸線 2 3 から離れる方向に向かって延びている。第 2 アーム 1 4 は、アーム基部 1 5 とアーム端部 1 6 とを含む。アーム基部 1 5 は、第 1 アーム 1 3 の先端部に接続されている。アーム端部 1 6 は、軸線 2 3 に交差（例えば直交）する軸線 2 4 まわりに回転するようにアーム基部 1 5 の先端部に接続されており、軸線 2 4 に沿ってアーム基部 1 5 から離れる方向に向かって延びている。

【 0 0 1 6 】

第 3 アーム 1 7 は、軸線 2 4 に交差（例えば直交）する軸線 2 5 まわりに揺動するようにアーム端部 1 6 の先端部に接続されている。先端部 1 8 は、軸線 2 5 に交差（例えば直交）する軸線 2 6 まわりに回転するように第 3 アーム 1 7 の先端部に接続されている。

【 0 0 1 7 】

このように、ロボット 3 は、基部 1 1 と旋回部 1 2 とを接続する関節 3 1 と、旋回部 1 2 と第 1 アーム 1 3 とを接続する関節 3 2 と、第 1 アーム 1 3 と第 2 アーム 1 4 とを接続する関節 3 3 と、第 2 アーム 1 4 においてアーム基部 1 5 とアーム端部 1 6 とを接続する関節 3 4 と、アーム端部 1 6 と第 3 アーム 1 7 とを接続する関節 3 5 と、第 3 アーム 1 7 と先端部 1 8 とを接続する関節 3 6 とを有する。

【 0 0 1 8 】

アクチュエータ 4 1 , 4 2 , 4 3 , 4 4 , 4 5 , 4 6 は、例えば電動モータ及び減速機を含み、関節 3 1 , 3 2 , 3 3 , 3 4 , 3 5 , 3 6 をそれぞれ駆動する。例えばアクチュエータ 4 1 は軸線 2 1 まわりに旋回部 1 2 を回転させ、アクチュエータ 4 2 は軸線 2 2 まわりに第 1 アーム 1 3 を揺動させ、アクチュエータ 4 3 は軸線 2 3 まわりに第 2 アーム 1 4 を揺動させ、アクチュエータ 4 4 は軸線 2 4 まわりにアーム端部 1 6 を回転させ、アクチュエータ 4 5 は軸線 2 5 まわりに第 3 アーム 1 7 を揺動させ、アクチュエータ 4 6 は軸線 2 6 まわりに先端部 1 8 を回転させる。

【 0 0 1 9 】

なお、ロボット 3 の具体的な構成は適宜変更可能である。例えばロボット 3 は、上記 6 軸の垂直多関節ロボットに更に 1 軸の関節を追加した 7 軸の冗長型ロボットであってもよく、所謂スカラー型の多関節ロボットであってもよい。

【 0 0 2 0 】

図 1 に戻り、制御システム C S 1 は、複数のロボット 3 を含む複数のマシンを制御する。制御システム C S 1 は、未定区間を含むロボット 3 の動作パスを表す複数のコマンドを順次呼び出すことと、未定区間に対する追加パスを生成することと、呼び出したコマンドと、追加パスとに基づいてロボット 3 を動作させることと、を実行するように構成されている。更に制御システム C S 1 は、1 のコマンドに基づいてロボット 3 を動作させている際に、周辺環境情報に基づいて追加パスを生成するように構成されている。

【 0 0 2 1 】

動作パスは、例えばロボット 3 の一部（例えば先端部 1 8 ）の移動経路を表す。動作パスは、先端部 1 8 の位置及び姿勢の推移を表してもよい。動作パスは、複数の区間を含む。複数の区間のそれぞれは、動作パスの 2 の経由点の間である。複数の区間は、複数の既定区間と、1 以上の未定区間とを含む。複数の既定区間のそれぞれは、2 の経由点の間の

10

20

30

40

50

移動経路（例えば先端部 18 の移動経路）が定められている区間である。1 以上の未定区間のそれぞれは、2 の経由点の間のパスが定められていない区間である。

【0022】

制御システムCS1は、周辺環境情報に基づいて未定区間に対する追加パスを生成する。追加パスは、未定区間における2の経由点の間のパスを表す。

【0023】

動作パスのうち、ワークの生産等の動作目的の要となる区間のみを予め教示して既定区間としておけば、残りの未定区間が制御システムCS1により補われるため、ロボット3に対する動作教示が容易になる。

【0024】

更に、制御システムCS1によれば、複数のコマンドに基づく動作をロボット3が開始した後の周辺環境情報に基づいて追加パスが生成され、生成された追加パスに基づきロボット3による動作が継続される。従って、周辺環境の変化に柔軟に対応した動作をロボット3に実行させることができる。

【0025】

以下、制御システムCS1の構成をより詳細に例示する。制御システムCS1は、複数のコントローラ100と、上位コントローラ200とを有する。複数のコントローラ100は、上位コントローラ200からの指令に基づいて、複数のロボット3をそれぞれ制御する。上位コントローラ200は、一連の作業を複数のロボット3に協働して遂行させるように、複数のコントローラ100のそれぞれに作業指令を送信する。複数のコントローラ100のそれぞれは、作業指令に基づいて、対応するロボット3を制御する。

【0026】

作業指令に基づくロボット3の制御において、コントローラ100は、複数のコマンドを順次呼び出すことと、1以上の未定区間のそれぞれに対する追加パスを生成することと、呼び出したコマンドと、追加パスとに基づいてロボット3を動作させることと、を実行する。コントローラ100は、1のコマンドに基づいてロボット3を動作させている際に、周辺環境情報に基づいて追加パスを生成する。

以下、コントローラ100と、上位コントローラ200と、シミュレーション装置300との構成をより詳細に例示する。

【0027】

〔コントローラ〕

図3は、コントローラ100の機能的な構成を例示するブロック図である。図3に示すように、コントローラ100は、機能上の構成要素（以下、「機能ブロック」という。）として、呼出部111と、パスプラン部112と、パス保存部113と、制御部114とを有する。

【0028】

呼出部111は、ロボット3の動作パスを表す複数のコマンドを順次呼び出す。複数のコマンドは、予めコントローラ100が記憶していてもよい。例えば、複数のコマンドを実行順に配列した動作プログラムが予めコントローラ100が記憶していてもよい。この場合、呼出部111は、コントローラ100が記憶する動作プログラムから複数のコマンドを順次フェッチする。フェッチとは、データを読み出すことを意味する。

【0029】

複数のコマンドは、クラウドからコントローラ100に送信されてもよい。この場合、呼出部111は、クラウドコンピュータからの受信データを一時的に記憶する受信バッファから複数のコマンドを順次フェッチすることとなる。

【0030】

パスプラン部112は、未定区間に対する追加パスを生成する。動作パスが複数の未定区間を含む場合、パスプラン部112は、複数の未定区間のそれぞれに対する追加パスを生成する。

【0031】

10

20

30

40

50

一例として、複数のコマンドは、ムーブコマンドと、オートコマンドとを含む。複数のコマンドは、複数のムーブコマンドと、1以上のオートコマンドとを含んでいてもよい。

【0032】

ムーブコマンドは、動作パスの経路点の情報を含む。経路点は、少なくとも先端部18の位置を定める。経路点は、先端部18の位置及び姿勢を定めてもよい。経路点は、先端部18の位置及び姿勢自体を定めるのに代えて、関節31, 32, 33, 34, 35, 36の角度を定めてもよい。関節31, 32, 33, 34, 35, 36の角度を定めることによっても、先端部18の位置及び姿勢が定まる。ムーブコマンドの経路点は、オペレータによるオフラインティーチング、又はオンラインティーチング等により教示された教示点であってもよい。

10

【0033】

ムーブコマンドは、パス特定情報を更に含んでもよい。パス特定情報は、ムーブコマンドの経路点までの区間のパスを特定し、その区間を上記既定区間とするための情報である。以下、ムーブコマンドの経路点までの区間を「ムーブ区間」といい、ムーブ区間のパスを「ムーブ区間パス」という。例えばパス情報は、ムーブ区間パスを一通りに定めるパス特定条件を表す。パス特定条件の具体例としては、ムーブ区間を直線により補間すること（線形補間）、ムーブ区間をS字状の曲線により補間すること（S字補間）等が挙げられる。

【0034】

1以上のオートコマンドのそれぞれは、未定区間の到達点となる動作パスの経路点の情報を含む。1以上のオートコマンドのそれぞれは、追加パスを生成する際の生成条件を表す条件情報を更に含んでもよい。生成条件の具体例としては、後述の追加コマンドに対するパス特定条件、追加パスにおける移動速度の条件、追加パスにおける加速度の条件、追加パスにおける減速度の条件、追加パスにおける先端部18の姿勢の条件、周辺環境情報に变化がない場合に生成済みの追加パスの再利用を許容するか否かの条件等が挙げられる。オートコマンドの経路点も、ムーブコマンドの経路点と同様に、オペレータによるオフラインティーチング、又はオンラインティーチング等により教示された教示点であってもよい。

20

【0035】

図4は、複数のコマンドを例示する図である。図4には、コマンドC1, C2, C3, C4, C5, C6, C7, C8, C9が示されている。コマンドC1, C2, C3, C4, C6, C7, C8は、ムーブコマンドであり、コマンドC5, C9はオートコマンドである。図4において、各コマンドの経路点はカッコ内に引数として入力される。

30

【0036】

コマンドC1, C2, C3, C4, C6, C7, C8において、「Move」の後に添えられた1のアルファベットがパス情報の一例に相当する。例えば、「L」は、始点における先端部18の位置及び姿勢と、終点における先端部18の位置及び姿勢との間を直線により補完することを示している。「S」は、始点における先端部18の位置及び姿勢と、終点における先端部18の位置及び姿勢との間をS字状の曲線により補完することを示している。「J」は、始点における関節31, 32, 33, 34, 35, 36の角度と、終点における関節31, 32, 33, 34, 35, 36の角度との間を直線により補完することを示している。

40

【0037】

コマンドC5, C9において、「Move」の後に添えられた「Auto」は、コマンドがオートコマンドであることを示している。

【0038】

図4のように、ムーブコマンドとオートコマンドとを含む複数のコマンドによれば、ムーブコマンドに対応する既定区間（ムーブ区間）と、オートコマンドに対応する未定区間とを含む動作パスが表される。図3に戻り、パスプラン部112は、オートコマンドに対応する未定区間に対する追加パスを生成する。例えばパスプラン部112は、オートコマ

50

ンドの経由点（到達点）経由点の直前の経由点（出発点）から、到達点までの追加パスを生成する。なお、オートコマンドが複数のコマンドの先頭にある場合等において、パスプラン部 1 1 2 は、ロボット 3 の先端部 1 8 の現在位置を出発点として追加パスを生成してもよい。

【 0 0 3 9 】

制御部 1 1 4 は、呼出部 1 1 1 が呼び出したコマンドと、追加パスとに基づいてロボット 3 を動作させる。例えば制御部 1 1 4 は、複数のムーブコマンドにそれぞれ対応する複数のムーブ区間パスと、1 以上のオートコマンドにそれぞれ対応する 1 以上の追加パスとを含む一連の動作パスに沿うようにロボット 3 を動作させる。

【 0 0 4 0 】

制御部 1 1 4 は、複数のムーブ区間パス及び 1 以上の追加パスのそれぞれに完全に沿わせるようにロボット 3 を動作させなくてもよく、少なくとも、複数のムーブ区間パス及び 1 以上の追加パスのそれぞれに部分的に沿わせるようにロボット 3 を動作させればよい。例えば制御部 1 1 4 は、複数のムーブ区間パス及び 1 以上の追加パスのそれぞれに少なくとも部分的には沿いつつも、動作パスの 1 以上の経由点を通過しないようにロボット 3 を動作させてもよい。

【 0 0 4 1 】

パスプラン部 1 1 2 は、1 のコマンドに基づいて制御部 1 1 4 がロボット 3 を動作させている際に、ロボット 3 の周辺環境情報に基づいて追加パスを生成する。例えばパスプラン部 1 1 2 は、ムーブコマンドに基づき制御部 1 1 4 がロボット 3 を動作させている際に、ムーブコマンドより後のオートコマンドと周辺環境情報とに基づいて、オートコマンドに対応する未定区間における到達点までの追加パスを生成する。例えばパスプラン部 1 1 2 は、ムーブコマンドに対応するムーブ区間パスに沿って制御部 1 1 4 がロボット 3 を動作させている際に、追加パスを生成する。パスプラン部 1 1 2 は、オートコマンドの 2 以上前のムーブコマンドに対応するムーブ区間パスに沿って制御部 1 1 4 がロボット 3 を動作させている際に、追加パスを生成してもよい。

【 0 0 4 2 】

パスプラン部 1 1 2 は、先行するオートコマンドに基づき制御部 1 1 4 がロボット 3 を動作させている際に、後続するオートコマンドと周辺環境情報とに基づいて、後続するオートコマンドに対応する未定区間に対する追加パスを生成してもよい。パスプラン部 1 1 2 は、後続するオートコマンドの 2 以上前のオートコマンドに基づき制御部 1 1 4 がロボット 3 を動作させている際に、後続するオートコマンドと周辺環境情報とに基づいて、後続するオートコマンドに対応する未定区間に対する追加パスを生成してもよい。

【 0 0 4 3 】

パスプラン部 1 1 2 は、周辺環境情報に基づいて、追加パスを定める 2 以上の新たなムーブコマンドを生成してもよい。以下、追加パスを定める 2 以上の新たなムーブコマンドを、「2 以上の追加コマンド」という。この場合、2 以上の追加コマンドにそれぞれ対応する 2 以上のムーブ区間パスが連なったパスが、追加パスとなる。

【 0 0 4 4 】

例えば図 5 に示すように、パスプラン部 1 1 2 は、モデルデータベース 1 5 1 と、干渉チェック部 1 5 2 と、パス生成部 1 5 3 とを有する。モデルデータベース 1 5 1 は、コンベヤ 2 及び複数のロボット 3 のモデル情報を記憶する。モデルデータベース 1 5 1 は、コンベヤ 2 及び複数のロボット 3 の周辺物体のモデル情報を更に記憶していてもよい。モデル情報は、構造及び大きさを特定する数値情報を含む。

【 0 0 4 5 】

干渉チェック部 1 5 2 は、コンベヤ 2 及び複数のロボット 3 の動作をシミュレートして、ロボット 3 が周辺環境に存在する周辺物体及びロボット自体に干渉しないかを確認する。例えば干渉チェック部 1 5 2 は、モデルデータベース 1 5 1 が記憶するモデル情報と、環境情報データベース 2 1 2 が記憶する周辺環境情報とに基づいて追加パスに基づくロボット 3 の動作をシミュレートして、ロボット 3 が周辺物体に干渉しないかを確認する。干

10

20

30

40

50

渉とは、シミュレーション空間において、ロボット3が周辺物体と重なることを意味する。シミュレーション空間において干渉が生じる場合、実空間においてはロボット3と周辺物体との衝突が生じることとなる。

【0046】

干渉チェック部152は、2以上の追加コマンドを含む2以上のコマンドに基づき制御部114が生成する一連の制御指令を算出し、算出した一連の制御指令に基づいてロボット3の動作をシミュレートしてもよい。

【0047】

パス生成部153は、追加パスを生成する。例えばパス生成部153は、まず出発点と到達点とを直線により補間して追加パスを仮生成し、仮生成した追加パスに基づくロボット3の動作を干渉チェック部152にシミュレートさせる。干渉チェック部152によるシミュレートの結果、ロボット3と周辺物体との干渉があると判定された場合、パス生成部153は、周辺物体と干渉しない経路点をランダムに生成し、出発点と到達点との間に追加する。以後、出発点と、生成した1以上の経路点と、到達点とを結ぶ追加パスによりロボット3と周辺物体との干渉が生じなくなるまで、経路点の生成と追加を繰り返す。その後、パス生成部153は、追加した1以上の経路点と、到達点とをそれぞれ経路点とする2以上の追加コマンドを生成する。

【0048】

このように、パスプラン部112が生成した追加パスのうち、ロボット3が周辺物体に干渉しないことが干渉チェック部152により確認される。このため、制御部114は、ロボット3が周辺物体に干渉しない場合に、パスプラン部112が生成した追加パスに基づいてロボット3を動作させることとなる。

【0049】

図3に戻り、パスプラン部112は、生成した2以上の追加パスをパス保存部113に記憶させる。例えばパスプラン部112は、生成した2以上の追加コマンドをパス保存部113に記憶させる。パスプラン部112は、コントローラ100と通信可能なシミュレーション装置に追加パスを生成させてもよい。例えばパスプラン部112は、出発点と到達点とを指定して、追加パスの生成をシミュレーション装置に要求してもよい。追加パスの生成の要求を受けたシミュレーション装置は、周辺環境情報に基づいて追加パスを生成する。このように、周辺環境情報に基づく追加パスの生成を、他の装置に実行させることも、周辺環境情報に基づいて追加パスを生成することに含まれる。

【0050】

パス保存部113が2以上の追加コマンドを記憶すると、呼出部111が2以上の追加コマンドを順次呼び出す。制御部114は、呼出部111が呼び出した2以上の追加コマンドに基づいてロボット3を動作させる。例えば制御部114は、2以上の追加コマンドにそれぞれ対応する2以上のムーブ区間パスのそれぞれに、少なくとも部分的に沿わせるようにロボット3を動作させる。

【0051】

パスプラン部112は、呼出部111がオートコマンドを呼び出した後に、オートコマンドと周辺環境情報とに基づいて、2以上の追加コマンドを生成してもよい。例えばパスプラン部112は、呼出部111が呼び出したオートコマンドの経路点（到達点）の直前の経路点（開始点）から到達点までの追加パスを生成する。

【0052】

呼出部111は、呼び出した2以上のコマンドをコマンド保存部115に記憶させてもよい。呼出部111は、2以上の追加コマンドを含む2以上のコマンドをコマンド保存部115に記憶させてもよい。例えば呼出部111は、オートコマンドを呼び出した場合には、オートコマンドに基づきパスプラン部112がパス保存部113に記憶させた2以上の追加コマンドをパス保存部113から呼び出してコマンド保存部115に記憶させてもよい。

【0053】

10

20

30

40

50

制御部 114 は、コマンド保存部 115 が記憶する 2 以上のコマンドに基づいてロボット 3 を動作させてもよい。例えば制御部 114 は、コマンド保存部 115 が記憶する 2 以上のコマンドに基づいてロボット 3 の一連の制御指令を生成し、一連の制御指令に基づいてロボット 3 を動作させてもよい。例えば制御部 114 は、コマンド保存部 115 が記憶する 2 以上のコマンドに基づいて、加減速を含む一連の速度パターンを生成し、一連の速度パターンに基づいてロボット 3 を動作させてもよい。例えば制御部 114 は、先端部 18 の位置及び姿勢に対する一連の速度パターンに基づく逆運動学演算により、関節 31, 32, 33, 34, 35, 36 の目標角度を算出することと、関節 31, 32, 33, 34, 35, 36 の角度を目標角度に追従させることを、所定の制御サイクルで繰り返し実行する。

10

【0054】

パスプラン部 112 は、オートコマンドの直前のムーブコマンドに対応する動作が完了する前に追加パスの生成が完了するタイミングで、追加パスの生成を開始してもよい。例えばパスプラン部 112 は、オートコマンドの直前のムーブコマンドに対応する動作が完了する前に、2 以上の追加コマンドをパス保存部 113 に格納し得るタイミングで、追加パスを生成してもよい。以下、オートコマンドの直前のムーブコマンドに対応する動作を、「先行動作」という。

【0055】

例えばパスプラン部 112 は、先行動作の完了予測タイミングから所定の生成余裕時間前のタイミングにて、追加パスの生成を開始してもよい。生成余裕時間は、追加パスの生成に要する時間以上となるように定められている。生成余裕時間は、追加パスの生成に要する時間と、上記制御指令の生成に要する時間との合計時間以上となるように定められている。

20

【0056】

先行動作の完了予測タイミングの具体例としては、先行動作からオートコマンドに基づく動作に移行するタイミング、先行動作からオートコマンドに基づく動作に移行する前の減速開始タイミング、等が挙げられる。

【0057】

呼出部 111 は、コマンド保存部 115 が記憶する 2 以上のコマンドに対応する動作の予想時間が、少なくともパスプラン部 112 による追加パスの生成に要する時間よりも長くなるように、コマンド保存部 115 に記憶させるコマンドの数を変更してもよい。例えば呼出部 111 は、上記予想時間が追加パスの生成に要する時間よりも短い場合に、コマンドの呼び出しサイクルを短くして、コマンド保存部 115 が記憶するコマンドの数を増やしてもよい。

30

【0058】

図 6 は、ロボットの動作期間と、追加パスの生成期間との関係を表すタイミングチャートであり、横軸が時間の経過を表す。図 6 に示すように、パスプラン部 112 は先行動作の完了予測タイミング t_1 よりも、生成余裕時間 T_{11} 前の開始タイミング t_2 において、追加パスの生成を開始する。生成余裕時間 T_{11} は、追加パスの生成時間 T_{13} 以上である。このため、生成余裕時間 T_{11} において追加パスの生成を開始すれば、先行動作が完了する完了予測タイミング t_1 までに追加パスの生成を完了させることができる。

40

【0059】

図 6 においては、コマンド保存部 115 が記憶する 2 以上のコマンドに対応する動作の予想時間 T_{12} が、生成余裕時間 T_{11} よりも長い。このため、オートコマンドがフェッチされた後、先行動作が完了するまでの時間に余裕があり、完了予測タイミング t_1 よりも生成余裕時間 T_{11} 前の開始タイミング t_2 において追加パスの生成を開始することができる。

【0060】

仮に、予想時間 T_{12} が生成余裕時間 T_{11} よりも短い場合には、オートコマンドがフェッチされるタイミングで、完了予測タイミング t_1 よりも生成余裕時間 T_{11} 前の開始

50

タイミング t_2 が既に過去の時間となるため、完了予測タイミング t_1 よりも前に追加パスの生成時間を十分に確保することができない。このような状況を回避するために、呼出部 111 は、予想時間 T_{12} が生成余裕時間 T_{11} よりも長くなるように、コマンド保存部 115 に記憶させるコマンドの数を変更する。

【0061】

図 3 に戻り、オートコマンドが上記条件情報を含む場合、パスプラン部 112 は、条件情報に更に基づいて追加パスを生成してもよい。例えばパスプラン部 112 は、条件情報が表す生成条件を追加パスが満たすように、周辺環境情報に基づく追加パスの生成を実行してもよい。

【0062】

複数のコマンドは、ムーブコマンド及びオートコマンドの他に、経路点をシフトさせるシフトコマンドを含んでもよい。パスプラン部 112 は、呼出部 111 が、シフトコマンドの後にオートコマンドを呼び出した場合に、オートコマンドの経路点（到達点）をシフトコマンドに基づきシフトさせたシフト到達点までの追加パスを生成する。複数のコマンドは、シフトコマンドによる経路点のシフトをオフにするシフトオフコマンドを更に含んでもよい。パスプラン部は、呼出部 111 が、シフトコマンドの後、且つシフトオフコマンドの前にオートコマンドを呼び出した場合に、オートコマンドの到達点をシフトコマンドに基づきシフトさせたシフト到達点までの追加パスを生成してもよい。

【0063】

コントローラ 100 は、タスク保存部 121 と、タスク選択部 123 とを更に備えてもよい。タスク保存部 121 は、2 以上のムーブコマンドをそれぞれ含む複数のタスクを保存する。タスク選択部 123 は、タスク保存部 121 に保存された複数のタスクのうち、1 のタスクを選択する。

【0064】

コントローラ 100 がタスク保存部 121 及びタスク選択部 123 を備える場合、呼出部 111 は、タスク選択部 123 が選択した 1 のタスクから 2 以上のコマンドを順次呼び出してもよい。呼出部 111 は、タスク選択部 123 が選択した 1 のタスクに含まれるムーブコマンドを呼び出す前に、オートコマンドを呼び出してもよい。例えば呼出部 111 は、タスク選択部 123 が選択した 1 のタスクの最初のムーブコマンドを呼び出す前に、オートコマンドを呼び出してもよい。タスク選択部 123 は、先に選択したタスクに含まれる 2 以上のコマンドが呼出部 111 により呼び出されたタイミングで、次のタスクを選択してもよい。

【0065】

タスク保存部 121 は、予め複数のタスクを保存していてもよいし、必要に応じ上位装置（例えばクラウドコンピュータ等）から複数のタスクを取得し、取得した複数のタスクを保存してもよい。タスク選択部 123 は、タスク保存部 121 が保存する複数のタスクから、予め定められた順序で 1 のタスクを選択してもよく、ロボット 3 の周辺環境情報に基づいて、周辺環境に適したタスクを自律的に選択してもよい。

【0066】

タスク選択部 123 が、複数のタスクから予め定められた順序で 1 のタスクを選択する場合、コントローラ 100 はフロー情報取得部 127 と、フロー保存部 122 とを更に有してもよい。フロー情報取得部 127 は、ユーザインタフェースへのユーザ入力等に基づいて、複数のタスクの実行順序を取得し、フロー保存部 122 に記憶させる。タスク選択部 123 は、フロー保存部 122 が記憶する実行順序、複数のタスクから 1 のタスクを選択する。

【0067】

複数のタスクの少なくともいずれかは、2 以上のムーブコマンドの前にオートコマンドを含んでもよい。例えば、複数のタスクのそれぞれが、2 以上のムーブコマンドの前にオートコマンドを含んでもよい。図 7 は、タスク選択部 123 の記憶内容を例示する模式図である。図 7 においては、タスク選択部 123 が複数のタスク 141 を記憶して

10

20

30

40

50

いる。複数のタスク 1 4 1 のそれぞれは、2 以上のムーブコマンドと、オートコマンドとを含んでおり、オートコマンドは 2 以上のムーブコマンドの前（例えば先頭）に配置されている。このように、複数のタスクのそれぞれがオートコマンドを含む場合、呼出部 1 1 1 は、タスク選択部 1 2 3 が選択した 1 のタスクからオートコマンドを呼び出す。

【0068】

図 3 に戻り、コントローラ 1 0 0 は、オートコマンド配置部 1 2 4 を更に有してもよい。オートコマンド配置部 1 2 4 は、1 のタスクにおいて最初の経路点がムーブコマンドにより定められている場合に、最初の経路点を定めるムーブコマンドをオートコマンドに置き換える。オートコマンド配置部 1 2 4 は、最初の経路点を定めるムーブコマンドの前にオートコマンドを挿入してもよい。例えば、オートコマンド配置部 1 2 4 は、最初の経路点を定めるムーブコマンドの前に、最初の経路点を定めるオートコマンドを挿入してもよい。オートコマンド配置部 1 2 4 によれば、オートコマンドを含まない既存のタスクを、オートコマンドを含むタスクに変換して活用することができる。

10

【0069】

図 8 に示すように、コントローラ 1 0 0 は、コマンド生成部 1 3 1 と、タスク生成部 1 3 2 とを更に有してもよい。コマンド生成部 1 3 1 は、2 以上のムーブコマンドを指定するタスク情報に基づいて、2 以上のムーブコマンドの前に実行されるオートコマンドを生成する。例えばコマンド生成部 1 3 1 は、タスク情報をユーザインタフェースへのユーザ入力に基づき取得する。例えばコマンド生成部 1 3 1 は、2 以上のムーブコマンドに基づいて、2 以上のムーブコマンドに対応する動作への移行動作における到達点（タスクの開始点）を特定し特定した到達点の情報を含むオートコマンドを生成する。

20

【0070】

タスク生成部 1 3 2 は、生成されたオートコマンドと、タスク情報とに基づいてタスクを生成する。例えばタスク生成部 1 3 2 は、タスク情報が指定する 2 以上のムーブコマンドの前に、コマンド生成部 1 3 1 が生成したオートコマンドを挿入してタスクを生成する。コマンド生成部 1 3 1 及びタスク生成部 1 3 2 によれば、オートコマンドを含むタスクの生成作業を簡素化することができる。また、オートコマンドの配置漏れに伴うタスク間の動作不良を防ぐことも可能となる。

【0071】

呼出部 1 1 1 は、1 のタスクにおいて最初の経路点がムーブコマンドにより定められている場合に、最初の経路点を定めるムーブコマンドを、最初の経路点を到達点とするオートコマンドに読み替えるように構成されていてもよい。この場合、オートコマンドを含まない既存のタスクを書き換えることなく残しつつ、1 のタスクの前に追加パスを補うことができる。

30

【0072】

図 3 に戻り、コントローラ 1 0 0 は、環境変化チェック部 1 2 5 を更に有してもよい。環境変化チェック部 1 2 5 は、周辺環境情報に基づいて、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がないかを確認する。例えば環境変化チェック部 1 2 5 は、上位コントローラ 2 0 0 の環境情報データベース 2 1 2（後述）が記憶する周辺環境情報に基づいて、周辺環境情報の変化がないかを確認する。

40

【0073】

一例として、環境変化チェック部 1 2 5 は、周辺環境情報の変化の有無を環境フラグによって表す。例えば環境変化チェック部 1 2 5 は、パスプラン部 1 1 2 が追加パスを生成する際に、環境フラグを「オフ」にし、環境情報データベース 2 1 2 から周辺環境情報を取得する。以下、このタイミングで取得する周辺環境情報を「基準情報」という。その後、環境変化チェック部 1 2 5 は、環境情報データベース 2 1 2 における周辺環境情報が更新されると、更新後の周辺環境情報と基準情報とを比較し、更新後の周辺環境情報と基準情報との間に差異を認識した場合に、環境フラグを「オフ」から「オン」に変更する。

【0074】

なお、周辺環境情報には、追加パスに沿ったロボット 3 の動作に影響を及ぼさない項目

50

も含まれ得る。環境変化チェック部 1 2 5 は、周辺環境情報のうち、追加パスに沿ったロボット 3 の動作に影響を及ぼし得る項目として予め指定された項目のみに基づいて、周辺環境情報の変化の有無を確認してもよい。

【 0 0 7 5 】

制御部 1 1 4 は、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がない場合に、追加パスに基づいてロボット 3 を動作させてもよい。例えば制御部 1 1 4 は、追加パスに基づいてロボット 3 を動作させる前に、環境フラグを確認し、環境フラグが「オフ」である場合に、追加パスに基づいてロボット 3 を動作させてもよい。

【 0 0 7 6 】

制御部 1 1 4 は、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がある場合に、追加パスに基づくロボット 3 の動作を中止させてもよい。例えば制御部 1 1 4 は、追加コマンドに基づいてロボット 3 を動作させる前に環境フラグを確認し、環境フラグが「オン」である場合に、コマンド保存部 1 1 5 の内容をクリア（消去）してもよい。これにより、生成済みの速度パターンに続く速度パターンが生成されなくなるので、生成済みの速度パターンに基づく動作が完了した時点で、ロボット 3 の動作は一時停止する場合もある。

【 0 0 7 7 】

パスプラン部 1 1 2 は、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がある場合に、変化後の周辺環境情報に基づいて追加パスを再生成してもよい。例えばパスプラン部 1 1 2 は、環境フラグが「オン」であることを制御部 1 1 4 が認識した場合に、追加コマンドに対応するオートコマンドと、変化後の周辺環境情報とに基づいて追加パスを再生成してもよい。例えばパスプラン部 1 1 2 は、再生成した 2 以上の追加コマンドをパス保存部 1 1 3 に記憶させる。その後、再生成された 2 以上の追加コマンドを呼出部 1 1 1 が順次読み出してコマンド保存部 1 1 5 に記憶させる。クリアされたコマンド保存部 1 1 5 に、再生成された 2 以上の追加コマンドが格納されると、ロボット 3 の動作が再開される。

【 0 0 7 8 】

なお、追加パスの生成は、少なくとも複数のコマンドに基づく動作をロボット 3 が開始した後に行われればよい。パスプラン部 1 1 2 は、オートコマンドの経由点の一つ前の経由点が発達したタイミングで直ちに追加パスを生成してもよい。

【 0 0 7 9 】

オートコマンドの経由点の一つ前の経由点が発達したタイミングで直ちに追加パスを生成する場合、追加パスの生成から、追加パスに基づくロボット 3 の動作開始までの待ち時間が長くなる可能性がある。待ち時間以内に追加パスが再生成される場合、ロボット 3 の動作を停止させることなく、再生成された追加パスによってロボット 3 の動作を継続させることができる。待ち時間に余裕がある場合に、環境変化チェック部 1 2 5 は、追加パスが再生成される度に周辺環境情報の変化の有無を確認し、パスプラン部 1 1 2 は周辺環境情報が変化する度に追加パスを再生成してもよい。

【 0 0 8 0 】

コントローラ 1 0 0 は、ステータス送信部 1 2 6 を更に有してもよい。ステータス送信部 1 2 6 は、制御部 1 1 4 によるロボット 3 の制御結果に基づいて、ロボット 3 のステータスを表すステータス情報を上位コントローラ 2 0 0 に送信する。ステータス情報は、例えば関節 3 1 , 3 2 , 3 3 , 3 4 , 3 5 , 3 6 の現在の角度を含む。

【 0 0 8 1 】

〔上位コントローラ〕

図 9 は、上位コントローラ 2 0 0 の構成を例示するブロック図である。図 9 に示すように、上位コントローラ 2 0 0 は、機能ブロックとして、情報収集部 2 1 1 と、環境情報データベース 2 1 2 と、システム制御部 2 1 3 と、プログラム保存部 2 1 4 とを有する。情報収集部 2 1 1 は、周辺環境情報を収集して環境情報データベース 2 1 2 に記憶させる。

【 0 0 8 2 】

10

20

30

40

50

情報収集部 2 1 1 は、少なくともパスプラン部 1 1 2 が追加パスを生成する前後に、周辺環境情報を更新する。例えば情報収集部 2 1 1 は、所定の更新サイクルにて周辺環境情報を繰り返し収集し、収集結果を時系列で環境情報データベース 2 1 2 に蓄積する。例えば情報収集部 2 1 1 は、コントローラ 1 0 0 のステータス送信部 1 2 6 から受信したステータス情報に基づいて周辺環境情報を収集してもよく、更にカメラなどの環境センサ 4 (図 1 参照) に基づいて周辺環境情報を収集してもよい。

【 0 0 8 3 】

プログラム保存部 2 1 4 は、一連の作業を複数のロボット 3 に協働して遂行させるように予め定められたシステムプログラムを記憶する。例えばシステムプログラムは、複数のロボット 3 のそれぞれに対する一連の作業指令を含む。システムプログラムは、一連の作業指令の少なくともいずれかについての出力条件を含んでいてもよい。

10

【 0 0 8 4 】

システム制御部 2 1 3 は、一連の作業を複数のロボット 3 に協働して遂行させるように、複数のロボット 3 のそれぞれに作業指令を出力する。例えばシステム制御部 2 1 3 は、システムプログラムに基づいて一連の作業指令を複数のロボット 3 のそれぞれに順次出力する。一連の作業指令のいずれかについて出力条件が定められている場合、システム制御部 2 1 3 は、環境情報データベース 2 1 2 の周辺環境情報が出力条件を満たした場合に、出力条件に対応する作業指令を出力する。

【 0 0 8 5 】

作業指令の具体例としては、上述した複数のタスクのいずれか一つ実行指令、フロー保存部 1 2 2 が記憶する実行順序による複数のタスクの実行指令等が挙げられる。

20

【 0 0 8 6 】

図 1 0 は、コントローラ 1 0 0 及び上位コントローラ 2 0 0 のハードウェア構成を例示するブロック図である。コントローラ 1 0 0 は、回路 1 9 0 を有する。回路 1 9 0 は、1 以上のプロセッサ 1 9 1 と、メモリ 1 9 2 と、ストレージ 1 9 3 と、通信ポート 1 9 4 と、ドライバ回路 1 9 5 とを有する。ストレージ 1 9 3 は、不揮発性の記憶媒体であり、複数のコマンドを順次呼び出すことと、追加パスを生成することと、呼び出したコマンドと、追加パスとに基づいてロボット 3 を動作させることと、を含み、1 のコマンドに基づいてロボット 3 を動作させている際に、周辺環境情報に基づいて追加パスを生成する制御方法をコントローラ 1 0 0 に実行させるためのプログラムを記憶する。例えばストレージ 1 9 3 は、上述した各機能ブロックをコントローラ 1 0 0 に構成させるためのプログラムを記憶する。ストレージ 1 9 3 は、フラッシュメモリ、又はハードディスク等の内蔵型の記憶媒体であってもよく、USBメモリ又は光ディスク等の可搬型の記憶媒体であってもよい。

30

【 0 0 8 7 】

メモリ 1 9 2 は、ストレージ 1 9 3 からロードされたプログラムを一時的に記憶する。メモリ 1 9 2 の具体例としては、ランダムアクセスメモリ等が挙げられる。1 以上のプロセッサ 1 9 1 は、メモリ 1 9 2 にロードされたプログラムを実行することにより、上述した各機能ブロックを構成する。1 以上のプロセッサ 1 9 1 は、演算結果を適宜メモリ 1 9 2 に記憶させる。

40

【 0 0 8 8 】

通信ポート 1 9 4 は、制御の同期通信用の通信ポートであり、1 以上のプロセッサ 1 9 1 からの要求に基づいて上位コントローラ 2 0 0 との間で通信を行う。ドライバ回路 1 9 5 は、1 以上のプロセッサ 1 9 1 からの要求に基づいて、関節 3 1 , 3 2 , 3 3 , 3 4 , 3 5 , 3 6 に駆動電力を供給する。ユーザインタフェース 1 9 6 は、1 以上のプロセッサ 2 9 1 からの要求に基づいて、ユーザとのコミュニケーションを行う。例えばユーザインタフェース 1 9 6 は、表示デバイスと、入力デバイスとを含む。表示デバイスの具体例としては、液晶モニタ又は有機 E L (E l e c t r o - L u m i n e s c e n c e) モニタ等が挙げられる。入力デバイスの具体例としては、キーボード、マウス、又はキーパッド等が挙げられる。入力デバイスは、所謂タッチパネルとして表示デバイスと一体化されて

50

いてもよい。

【 0 0 8 9 】

上位コントローラ 2 0 0 は、回路 2 9 0 を有する。回路 2 9 0 は、1 以上のプロセッサ 2 9 1 と、メモリ 2 9 2 と、ストレージ 2 9 3 と、通信ポート 2 9 4 と、入出力ポート 2 9 5 と、通信ポート 2 9 6 とを有する。ストレージ 2 9 3 は、不揮発性の記憶媒体であり、上述した各機能ブロックを上位コントローラ 2 0 0 に構成させるためのプログラムを記憶している。ストレージ 2 9 3 は、フラッシュメモリ、又はハードディスク等の内蔵型の記憶媒体であってもよく、U S B メモリ又は光ディスク等の可搬型の記憶媒体であってもよい。

【 0 0 9 0 】

メモリ 2 9 2 は、ストレージ 2 9 3 からロードされたプログラムを一時的に記憶する。メモリ 2 9 2 の具体例としては、ランダムアクセスメモリ等が挙げられる。1 以上のプロセッサ 2 9 1 は、メモリ 2 9 2 にロードされたプログラムを実行することにより、上述した各機能ブロックを構成する。1 以上のプロセッサ 2 9 1 は、演算結果を適宜メモリ 2 9 2 に記憶させる。

【 0 0 9 1 】

通信ポート 2 9 4 は、制御の同期通信用の通信ポートであり、1 以上のプロセッサ 2 9 1 からの要求に基づいてコントローラ 1 0 0 との間で通信を行う。入出力ポート 2 9 5 は、1 以上のプロセッサ 2 9 1 からの要求に基づいて、環境センサ 4 等からの情報を取得する。通信ポート 2 9 6 は、制御の同期通信とは別系統の通信用の通信ポートであり、1 以上のプロセッサ 2 9 1 からの要求に基づいて、シミュレーション装置 3 0 0 との間で同期通信を行う。このハードウェア構成において、コントローラ 1 0 0 は、上位コントローラ 2 0 0 を介してシミュレーション装置 3 0 0 と通信を行うこととなる。

【 0 0 9 2 】

以上に示したハードウェア構成はあくまで一例なので、適宜変更可能である。例えば、制御システム C S 1 は、必ずしもコントローラ 1 0 0 と、上位コントローラ 2 0 0 とに分かれていなくてもよい。例えばコントローラ 1 0 0 が上位コントローラ 2 0 0 に組み込まれていてもよい。また、制御システム C S 1 は、コントローラ 1 0 0 及び上位コントローラ 2 0 0 に加えて、上述のシミュレーション装置を更に備えていてもよい。この場合、コントローラ 1 0 0 は、周辺環境情報に基づく追加パスの生成をシミュレーション装置に実行させるように構成されていてもよい。

【 0 0 9 3 】

〔 制御手順 〕

続いて、制御方法の一例として、制御システム C S 1 が実行する制御手順を具体的に例示する。この手順は、複数のコマンドを順次呼び出すことと、追加パスを生成することと、コマンドと、追加パスとに基づいてロボット 3 を動作させることと、を含み、1 のコマンドに基づいてロボット 3 を動作させている際に、周辺環境情報に基づいて追加パスを生成する。制御システム C S 1 による制御手順は、コントローラ 1 0 0 が実行するタスク生成手順、オートコマンド配置手順、及びプログラムの生成手順と、上位コントローラ 2 0 0 が実行するシステム制御手順と、コントローラ 1 0 0 が実行するコマンド呼出手順、環境変化チェック手順、及びロボットの制御手順とを含み得る。以下、各手順を詳細に例示する。

【 0 0 9 4 】

（ オートコマンド配置手順 ）

図 1 1 に示すように、コントローラ 1 0 0 は、ステップ S 0 1 , S 0 2 , S 0 3 を実行する。ステップ S 0 1 では、オートコマンド配置部 1 2 4 が、タスク保存部 1 2 1 が記憶する複数のタスクから、オートコマンドを含まない 1 のタスクを選択する。ステップ S 0 2 では、オートコマンド配置部 1 2 4 が、1 のタスクにおいて最初の経由点がムーブコマンドにより定められている場合に、最初の経由点を定めるムーブコマンドをオートコマンドに置き換える。オートコマンド配置部 1 2 4 は、最初の経由点を定めるムーブコマンド

の前にオートコマンドを挿入してもよい。

【 0 0 9 5 】

ステップ S 0 3 では、オートコマンドを含まない全てのタスクに対してオートコマンドの配置が完了したか否かをオートコマンド配置部 1 2 4 が確認する。ステップ S 0 3 において、オートコマンドを含まないタスクが残っていると判定した場合、コントローラ 1 0 0 は処理をステップ S 0 1 に戻す。ステップ S 0 3 において、オートコマンドを含まない全てのタスクに対してオートコマンドの配置が完了したと判定した場合、コントローラ 1 0 0 は処理を完了する。

【 0 0 9 6 】

(タスク生成手順)

図 1 2 に示すように、コントローラ 1 0 0 は、ステップ S 1 1 , S 1 2 , S 1 3 , S 1 4 を実行する。ステップ S 1 1 では、コマンド生成部 1 3 1 が、2 以上のムーブコマンドを指定するタスク情報を取得する。コマンド生成部 1 3 1 は、ユーザインタフェース 1 9 6 からタスク情報を取得してもよい。

【 0 0 9 7 】

ステップ S 1 2 では、コマンド生成部 1 3 1 が、取得したタスク情報に基づいて、2 以上のムーブコマンドの前に実行されるオートコマンドを生成する。例えばコマンド生成部 1 3 1 は、2 以上のムーブコマンドに基づいて、2 以上のムーブコマンドに対応する動作への移行動作における到達点 (タスクの開始点) の情報を含むオートコマンドを生成する。

【 0 0 9 8 】

ステップ S 1 3 では、タスク生成部 1 3 2 が、生成されたオートコマンドと、タスク情報とに基づいてタスクを生成する。例えばタスク生成部 1 3 2 は、タスク情報が指定する 2 以上のムーブコマンドの前に、コマンド生成部 1 3 1 が生成したオートコマンドを挿入してタスクを生成する。ステップ S 1 4 では、タスク生成部 1 3 2 が、生成したタスクをタスク保存部 1 2 1 に記憶させる。以上でタスクの生成手順が完了する。

【 0 0 9 9 】

(プログラムの生成手順)

図 1 3 に示すように、コントローラ 1 0 0 は、ステップ S 2 1 , S 2 2 を実行する。ステップ S 2 1 では、フロー情報取得部 1 2 7 が、複数のタスクの実行順序を取得する。ステップ S 2 2 では、フロー情報取得部 1 2 7 が、実行順序をフロー保存部 1 2 2 に記憶させる。以上でロボット 3 の動作プログラムの生成が完了する。

【 0 1 0 0 】

(システム制御手順)

図 1 4 に示すように、上位コントローラ 2 0 0 は、ステップ S 3 1 , S 3 2 , S 3 3 を実行する。ステップ S 3 1 では、システム制御部 2 1 3 が、プログラム保存部 2 1 4 のシステムプログラムに基づいて、次にコントローラ 1 0 0 に出力すべき作業指令を特定する。以下、特定した作業指令を、出力予定の作業指令という。ステップ S 3 2 では、情報収集部 2 1 1 が、周辺環境情報を収集し、収集した 1 セットの周辺環境情報に時刻情報を付与して環境情報データベース 2 1 2 に記憶させる。ステップ S 3 3 では、出力予定の作業指令に出力条件が付与されているかをシステム制御部 2 1 3 が確認する。

【 0 1 0 1 】

ステップ S 3 3 において、出力予定の作業指令に出力条件が付与されていると判定した場合、上位コントローラ 2 0 0 はステップ S 3 4 を実行する。ステップ S 3 4 では、周辺環境情報が出力条件を満たしているか否かをシステム制御部 2 1 3 が確認する。ステップ S 3 4 において、周辺環境情報が出力条件を満たしていないと判定した場合、上位コントローラ 2 0 0 はステップ S 3 5 を実行する。ステップ S 3 5 では、情報収集部 2 1 1 が更新サイクルの経過を待機する。その後、上位コントローラ 2 0 0 は処理をステップ S 3 2 に戻す。以後、周辺環境情報が出力条件を満たすまでは、更新サイクルごとに周辺環境情報の収集が繰り返される。

【 0 1 0 2 】

10

20

30

40

50

ステップ S 3 4 において、周辺環境情報が出力条件を満たしていると判定した場合、上位コントローラ 2 0 0 はステップ S 3 6 , S 3 7 を実行する。ステップ S 3 3 において、出力予定の作業指令に出力条件が付与されていないと判定した場合も、上位コントローラ 2 0 0 はステップ S 3 6 , S 3 7 を実行する。ステップ S 3 6 では、システム制御部 2 1 3 が、出力予定の作業指令をコントローラ 1 0 0 に送信する。ステップ S 3 7 では、システム制御部 2 1 3 が更新サイクルの経過を待機する。その後、コントローラ 1 0 0 は処理をステップ S 3 1 に戻す。上位コントローラ 2 0 0 は以上の処理を繰り返し実行する。

【 0 1 0 3 】

(コマンド呼出手順)

図 1 5 に示すように、コントローラ 1 0 0 は、まずステップ S 4 1 , S 4 2 , S 4 3 を実行する。ステップ S 4 1 では、タスク選択部 1 2 3 が、タスク保存部 1 2 1 に保存された複数のタスクのうち、1 のタスクを選択する。例えばタスク選択部 1 2 3 は、フロー保存部 1 2 2 が記憶する実行順序、複数のタスクから 1 のタスクを選択する。ステップ S 4 2 では、呼出部 1 1 1 が、上記 1 のタスクから 1 のコマンドを呼び出す。ステップ S 4 3 では、上記 1 のコマンドがムーブコマンドであるか否かを呼出部 1 1 1 が確認する。

10

【 0 1 0 4 】

ステップ S 4 3 において、1 のコマンドがムーブコマンドではないと判定した場合、コントローラ 1 0 0 はステップ S 4 4 を実行する。ステップ S 4 4 では、上記 1 のコマンドがオートコマンドであるか否かを呼出部 1 1 1 が確認する。ステップ S 4 4 において、1 のコマンドがオートコマンドではないと判定した場合、コントローラ 1 0 0 は処理をステップ S 4 2 に戻す。

20

【 0 1 0 5 】

ステップ S 4 4 において、1 のコマンドがオートコマンドであると判定した場合、コントローラ 1 0 0 はステップ S 4 5 , S 4 6 , S 4 7 を実行する。ステップ S 4 5 では、パスプラン部 1 1 2 が、追加パスの生成開始タイミングを待機する。例えばパスプラン部 1 1 2 は、先行動作の完了予測タイミングから所定の生成余裕時間前のタイミングを待機する。ステップ S 4 6 では、パスプラン部 1 1 2 が、オートコマンドの到達点の直前の経由点 (出発点) から、オートコマンドの到達点までの追加パスを生成し、生成した追加パスをパス保存部 1 1 3 に記憶させる。ステップ S 4 7 では、環境変化チェック部 1 2 5 が、環境フラグを「オフ」にする (クリアする) 。

30

【 0 1 0 6 】

次に、コントローラ 1 0 0 はステップ S 5 3 を実行する。ステップ S 5 3 では、パスプラン部 1 1 2 が、呼出部 1 1 1 によるコマンドの呼出先を、タスク保存部 1 2 1 からパス保存部 1 1 3 に変更する。その後、コントローラ 1 0 0 は処理をステップ S 4 2 に戻す。ステップ S 4 2 においては、パス保存部 1 1 3 からコマンドが呼び出されることとなる。

【 0 1 0 7 】

ステップ S 4 3 において、1 のコマンドがムーブコマンドであると判定した場合、コントローラ 1 0 0 はステップ S 5 4 , S 5 5 を実行する。ステップ S 5 4 では、呼出部 1 1 1 が、呼び出したコマンドをコマンド保存部 1 1 5 に記憶させる。ステップ S 5 5 では、呼び出したコマンドが追加コマンドであるか否かを呼出部 1 1 1 が確認する。

40

【 0 1 0 8 】

ステップ S 5 5 において、呼び出したコマンドが追加コマンドであると判定した場合、コントローラ 1 0 0 はステップ S 5 6 を実行する。ステップ S 5 6 では、パス保存部 1 1 3 が記憶する全ての追加コマンドの呼出が完了したか否かを呼出部 1 1 1 が確認する。

【 0 1 0 9 】

ステップ S 5 6 において、全ての追加コマンドの呼出が完了したと判定した場合、コントローラ 1 0 0 はステップ S 5 7 を実行する。ステップ S 5 7 では、呼出部 1 1 1 が、コマンドの呼出先を、パス保存部 1 1 3 からタスク保存部 1 2 1 に変更する。

【 0 1 1 0 】

次に、コントローラ 1 0 0 はステップ S 5 8 を実行する。ステップ S 5 6 において、ま

50

だ呼び出していない追加コマンドが残っていると判定した場合、コントローラ 100 はステップ S 57 を実行することなくステップ S 58 を実行する。ステップ S 55 において、呼び出したコマンドが追加コマンドではないと判定した場合、コントローラ 100 はステップ S 56 , S 57 を実行することなくステップ S 58 を実行する。ステップ S 58 では、1 のタスクに含まれる全てのコマンドの呼出が完了したか否かを呼出部 111 が確認する。

【0111】

ステップ S 58 において、1 のタスクにまだ呼び出していないコマンドが残っていると判定した場合、コントローラ 100 は処理をステップ S 42 に戻す。これにより、1 のタスクからのコマンドの呼出が継続される。ステップ S 58 において、1 のタスクに含まれる全てのコマンドの呼出が完了したと判定した場合、コントローラ 100 は処理をステップ S 41 に戻す。これにより、次のタスクが選択される。コントローラ 100 は、以上の手順を繰り返し実行する。なお、図 15 は、複数のコマンドがムーブコマンド及びオートコマンドのいずれでもないコマンド（例えば上述のシフトコマンド）である場合の処理を省略している。複数のコマンドがムーブコマンド及びオートコマンドのいずれでもないコマンドを含む場合には、そのコマンドに対する処理が適宜追加される。

【0112】

図 16 は、ステップ S 46 において追加パスを生成する手順を例示するフローチャートである。図 16 に示すように、シミュレーション装置 300 は、まずステップ S 92 , S 93 を実行する。ステップ S 92 では、パスプラン部 112 が、上記出発点と到達点とを直線により補間して追加パスを仮生成する。ステップ S 93 では、追加パスに基づくロボット 3 の動作を干渉チェック部 152 がシミュレートし、ロボット 3 と周辺物体との干渉がないか否かを確認する。

【0113】

ステップ S 93 において、干渉があると判定した場合、コントローラ 100 はステップ S 94 を実行する。ステップ S 94 では、パスプラン部 112 が、周辺物体と干渉しない経路点をランダムに生成して出発点と到達点との間に挿入し、追加パスを修正する。その後、コントローラ 100 は処理をステップ S 93 に戻す。以後、追加パスによりロボット 3 と周辺物体との干渉が生じなくなるまで、経路点の生成と追加とが繰り返される。

【0114】

ステップ S 93 において、干渉がないと判定した場合、コントローラ 100 はステップ S 95 を実行する。ステップ S 95 では、パスプラン部 112 が、追加した 1 以上の経路点と、終点とをそれぞれ経路点とする複数の追加コマンドを生成する。以上でパスの生成手順が完了する。

【0115】

（環境変化チェック手順）

図 17 に示すように、コントローラ 100 は、ステップ S 61 , S 62 を実行する。ステップ S 61 では、環境変化チェック部 125 が、上記基準情報として、環境情報データベース 212 から周辺環境情報を取得する。ステップ S 62 では、環境変化チェック部 125 が、上述したコマンド呼出手順において環境フラグをクリアしたか否かを確認する。

【0116】

ステップ S 62 において、環境フラグをクリアしていないと判定した場合、コントローラ 100 は、ステップ S 63 を実行する。ステップ S 63 では、環境フラグの更新サイクルが経過したか否かを確認する。ステップ S 63 において、更新サイクルが経過していないと判定した場合、コントローラ 100 は処理をステップ S 62 に戻す。以後、環境フラグをクリアするか、更新サイクルが経過するまでは、ステップ S 62 , S 63 が繰り返される。

【0117】

ステップ S 63 において、更新サイクルが経過したと判定した場合、コントローラ 100 はステップ S 64 , S 65 を実行する。ステップ S 64 では、環境変化チェック部 12

10

20

30

40

50

5 が、環境情報データベース 212 から周辺環境情報を取得する。ステップ S 65 では、環境変化チェック部 125 が、ステップ S 64 で取得した周辺環境情報と、基準情報とを比較して、周辺環境情報に変化があるか否かを確認する。

【0118】

ステップ S 65 において、周辺環境情報に変化はないと判定した場合、コントローラ 100 は処理をステップ S 62 に戻す。ステップ S 65 において、周辺環境情報に変化があると判定した場合、コントローラ 100 はステップ S 66 を実行する。ステップ S 66 では、環境変化チェック部 125 が、環境フラグを「オフ」から「オン」に変更する。その後、コントローラ 100 は処理をステップ S 62 に戻す。

【0119】

ステップ S 62 において、環境フラグをクリアしたと判定した場合、コントローラ 100 はステップ S 67 を実行する。ステップ S 67 では、環境変化チェック部 125 が環境情報データベース 212 から周辺環境情報を取得し、取得した周辺環境情報によって基準情報を更新する。その後、コントローラ 100 は処理をステップ S 62 に戻す。以後、環境フラグのクリアに応じて基準情報を更新しつつ、更新サイクルごとに周辺環境情報の変化の有無を確認することが繰り返される。

【0120】

(ロボットの制御手順)

図 18 に示すように、コントローラ 100 は、まずステップ S 71 , S 72 を実行する。ステップ S 71 では、制御部 114 が、コマンド保存部 115 から 1 のコマンドを読み出す。ステップ S 72 では、1 のコマンドが追加コマンドであるか否かを制御部 114 が確認する。

【0121】

ステップ S 72 において、1 のコマンドが追加コマンドではないと判定した場合、コントローラ 100 はステップ S 74 , S 75 を実行する。ステップ S 74 では、制御部 114 が、1 のコマンドと、先に読み出した 1 以上のコマンドとを含む 2 以上のコマンドに基づく一連の制御指令 (例えば速度パターン) を生成する。ステップ S 75 では、制御部 114 が、ステップ S 74 で生成した一連の制御指令に基づくロボット 3 の制御を開始する。その後、コントローラ 100 は処理をステップ S 71 に戻す。

【0122】

ステップ S 72 において、1 のコマンドが追加コマンドであると判定した場合、コントローラ 100 はステップ S 73 を実行する。ステップ S 73 では、環境フラグがオフであるか否かを制御部 114 が確認する。ステップ S 73 において、環境フラグがオフであると判定した場合、コントローラ 100 は処理をステップ S 74 に進める。

【0123】

ステップ S 73 において、環境フラグがオンであると判定した場合、コントローラ 100 はステップ S 81 , S 82 , S 83 を実行する。ステップ S 81 では、制御部 114 が、コマンド保存部 115 の内容をクリア (消去) する。ステップ S 82 では、パスプラン部 112 が、周辺環境情報に基づいて、追加コマンドに対応するオートコマンドのタイミングを待機する。ステップ S 82 では、パスプラン部 112 が、オートコマンドの到達点の直前の経由点 (出発点) から、オートコマンドの到達点までの追加パスを再生成し、生成した追加パスをパス保存部 113 に記憶させる。ステップ S 83 では、環境変化チェック部 125 が、環境フラグを「オフ」にする (クリアする)。

【0124】

次に、コントローラ 100 はステップ S 86 を実行する。ステップ S 86 では、パスプラン部 112 が、呼出部 111 によるコマンドの呼出先を、タスク保存部 121 からパス保存部 113 に変更する。これにより、再生成された 2 以上の追加コマンドが、上述したコマンド呼出手順において呼出部 111 により順次呼び出され、コマンド保存部 115 に保存されることとなる。その後、コントローラ 100 は処理をステップ S 71 に戻す。コントローラ 100 は以上の処理を繰り返し実行する。

10

20

30

40

50

【 0 1 2 5 】

〔まとめ〕

ロボットシステム 1 は、未定区間を含むロボット 3 の動作パスを表す複数のコマンドを順次呼び出す呼出部 1 1 1 と、未定区間に対する追加パスを生成するパスプラン部 1 1 2 と、呼出部 1 1 1 が呼び出したコマンドと、追加パスとに基づいてロボット 3 を動作させる制御部 1 1 4 と、を備え、パスプラン部 1 1 2 は、1 のコマンドに基づいて制御部 1 1 4 がロボット 3 を動作させている際に、ロボット 3 の周辺環境情報に基づいて追加パスを生成する。

【 0 1 2 6 】

このロボットシステム 1 によれば、複数のコマンドに基づく動作をロボット 3 が開始した後の周辺環境情報に基づいて追加パスが生成され、生成された追加パスに基づきロボット 3 による動作が継続される。従って、周辺環境の変化、作業内容の変化、及び作業手順の変化等に対応した動作をロボット 3 に実行させることができる。以下、複数のコマンドに基づく動作をロボット 3 が開始した後に追加パスを生成することを、「オンラインパス生成」という。

【 0 1 2 7 】

複数のコマンドは、動作パスの経由点の情報を含むムーブコマンドと、未定区間の到達点となる動作パスの経由点の情報を含むオートコマンドと、を含み、パスプラン部 1 1 2 は、ムーブコマンドに基づき制御部 1 1 4 がロボット 3 を動作させている際に、ムーブコマンドより後のオートコマンドと周辺環境情報とに基づいて、未定区間における到達点までの追加パスを生成してもよい。予め定めた動作パスでロボット 3 を動作させる区間と、パスプラン部 1 1 2 が生成した追加パスでロボット 3 を動作させる区間とを容易に指定することが可能となる。既存のロボット 3 の動作プログラムは、ムーブコマンドを含む複数のコマンドの列挙により記述されている場合がある。オートコマンドと周辺環境情報とに基づいて追加パスを生成する構成によれば、既存の動作プログラムを利用して、オンラインパス生成を含む動作プログラムを容易に生成することができる。

【 0 1 2 8 】

2 以上のムーブコマンドをそれぞれ含む複数のタスクを保存するタスク保存部 1 2 1 と、タスク保存部 1 2 1 に保存された複数のタスクのうち、1 のタスクを選択するタスク選択部 1 2 3 と、を更に備え、呼出部 1 1 1 は、タスク選択部 1 2 3 が選択した 1 のタスクに含まれるムーブコマンドを呼び出す前に、オートコマンドを呼び出してよい。個別に生成された複数のタスクに基づいて、ロボット 3 を容易に動作させることができる。

【 0 1 2 9 】

複数のタスクの少なくともいずれかは、2 以上のムーブコマンドの前にオートコマンドを含み、呼出部 1 1 1 は、タスク選択部 1 2 3 が選択した 1 のタスクからオートコマンドを呼び出してよい。個別に生成された複数のタスクに基づいて、ロボット 3 を更に容易に動作させることができる。

【 0 1 3 0 】

2 以上のムーブコマンドを指定するタスク情報に基づいて、2 以上のムーブコマンドの前に実行されるオートコマンドを生成するコマンド生成部 1 3 1 と、生成されたオートコマンドと、タスク情報とに基づいてタスクを生成するタスク生成部 1 3 2 と、を更に備えてもよい。オートコマンドを含むタスクの生成作業を簡素化することができる。また、オートコマンドの配置漏れに伴うタスク間の動作不良を防ぐことも可能となる。

【 0 1 3 1 】

1 のタスクにおいて最初の経由点がムーブコマンドにより定められている場合に、最初の経由点を定めるムーブコマンドをオートコマンドに置き換えるか、最初の経由点を定めるムーブコマンドの前にオートコマンドを挿入する、オートコマンド配置部 1 2 4 を更に有してもよい。オートコマンドを有しない既存のタスクを、オートコマンドを含むタスクに容易に変換することができる。

【 0 1 3 2 】

10

20

30

40

50

呼出部 1 1 1 は、1 のタスクにおいて最初の経由点がムーブコマンドにより定められている場合に、最初の経由点を定めるムーブコマンドを、最初の経由点を到達点とするオートコマンドに読み替えてもよい。オートコマンドを含まない既存のタスクを書き換えることなく残しつつ、オンラインパス生成を利用することができる。

【0 1 3 3】

パスプラン部 1 1 2 は、到達点の直前の経由点から到達点までの追加パスを生成してもよい。直前の経由点を出発点とすることで、オートコマンドで出発点を定める必要がなくなるので、オートコマンドを簡素化することができる。

【0 1 3 4】

複数のコマンドは、経由点をシフトさせるシフトコマンドを含み、パスプラン部 1 1 2 は、呼出部 1 1 1 が、シフトコマンドの後にオートコマンドを呼び出した場合に、オートコマンドの経由点をシフトコマンドに基づきシフトさせたシフト到達点までの追加パスを生成してもよい。シフトコマンドをオートコマンドにも適用可能とすることで、動作プログラミングの更なる簡素化を図ることができる。

【0 1 3 5】

パスプラン部 1 1 2 は、オートコマンドの直前のムーブコマンドに対応する動作が完了する前に追加パスの生成が完了するタイミングで、追加パスの生成を開始してもよい。この場合、追加パスの生成を待つためにロボット 3 が停止する状態を減らし、ロボット 3 をよりスムーズに動作させることができる。

【0 1 3 6】

パスプラン部 1 1 2 は、周辺環境情報に基づいて、追加パスを定める 2 以上の新たなムーブコマンドを生成してもよい。追加パスの生成のための演算を簡素化することができる。

【0 1 3 7】

パスプラン部 1 1 2 は、呼出部 1 1 1 がオートコマンドを呼び出した後に、オートコマンドと周辺環境情報とに基づいて、追加パスを定める 2 以上の新たなムーブコマンドを生成し、呼出部 1 1 1 は、2 以上の新たなムーブコマンドを含む 2 以上のコマンドをコマンド保存部に記憶させ、制御部 1 1 4 は、コマンド保存部が記憶する 2 以上のコマンドに基づいてロボット 3 を動作させてもよい。コマンド保存部が記憶する 2 以上のコマンドに基づいてロボット 3 を動作させることによって、次のコマンドを待つためにロボット 3 が停止する状態を減らし、ロボット 3 をよりスムーズに動作させることができる。このように、コマンド保存部が記憶する 2 以上のコマンドに基づきロボット 3 を動作させるしくみに、オンラインパス生成の結果を組み合わせることによって、ムーブコマンドのパス情報に基づく動作区間と、追加パスに基づく動作区間とを滑らかにつなぎ合わせ、ロボット 3 をよりスムーズに動作させることができる。

【0 1 3 8】

制御部 1 1 4 は、コマンド保存部が記憶する 2 以上のコマンドに基づいてロボット 3 の一連の制御指令を生成し、一連の制御指令に基づいてロボット 3 を動作させてもよい。ロボット 3 をよりスムーズに動作させることができる。

【0 1 3 9】

呼出部 1 1 1 は、コマンド保存部が記憶する 2 以上のコマンドに対応する動作の予想時間が、少なくともパスプラン部 1 1 2 による追加パスの生成に要する時間よりも長くなるように、コマンド保存部に記憶させるコマンドの数を変更してもよい。追加パスの生成を待つためにロボット 3 が停止する状態を更に減らし、ロボット 3 をよりスムーズに動作させることができる。

【0 1 4 0】

オートコマンドは、周辺環境情報に基づいて前記追加パスを生成する際の生成条件を表す条件情報を含み、パスプラン部 1 1 2 は、条件情報に更に基づいて追加パスを生成してもよい。条件情報の設定によって、より適切な追加パスを生成することができる。

【0 1 4 1】

パスプラン部 1 1 2 が生成した追加パスに基づくロボット 3 の動作をシミュレートして

10

20

30

40

50

、ロボット 3 が周辺環境に存在する周辺物体に干渉しないか確認する干渉チェック部 3 1 2 を更に有し、制御部 1 1 4 は、ロボット 3 が周辺物体に干渉しない場合に、パスプラン部 1 1 2 が生成した追加パスに基づいてロボット 3 を動作させてもよい。オンラインパス生成を含むロボット 3 の動作の信頼性を向上させることができる。

【0 1 4 2】

少なくとも、パスプラン部 1 1 2 が追加パスを生成する前後に、周辺環境情報を更新する情報収集部 2 1 1 と、周辺環境情報に基づいて、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がないかを確認する環境変化チェック部 1 2 5 と、を更に備えてもよい。オンラインパス生成を含むロボット 3 の動作の信頼性を更に向上させることができる。

10

【0 1 4 3】

制御部 1 1 4 は、パスプラン部 1 1 2 が追加パスを生成する前後における周辺環境情報の変化がない場合に、追加パスに基づいてロボット 3 を動作させてもよい。オンラインパス生成を含むロボット 3 の動作の信頼性を更に向上させることができる。

【0 1 4 4】

パスプラン部 1 1 2 は、追加パスを生成する前後における周辺環境情報の変化がある場合に、変化後の周辺環境情報に基づいて追加パスを再生成してもよい。オンラインパス生成を含むロボット 3 の動作の信頼性を更に向上させることができる。

【0 1 4 5】

上述の実施形態は、未定区間を含むロボット 3 の動作パスの経路点の情報を含む複数のムーブコマンドと、未定区間の到達点となる動作パスの経路点の情報を含むオートコマンドと、を含む複数のコマンドを順次呼び出す呼出部 1 1 1 と、オートコマンドと、ロボット 3 の周辺環境情報とに基づいて、未定区間における到達点までの追加パスを生成するパスプラン部 1 1 2 と、呼出部 1 1 1 が呼び出したコマンドと、追加パスとに基づいてロボット 3 を動作させる制御部 1 1 4 と、を備えるロボットシステムを含む。

20

【0 1 4 6】

更に上述の実施形態は、ロボット 3 の動作パスの経路点の情報をそれぞれが含む 2 以上のムーブコマンドを指定するタスク情報に基づいて、2 以上のムーブコマンドに対応する動作への移行動作における到達点の情報を含むオートコマンドを生成するコマンド生成部 1 3 1 と、生成したオートコマンドと、タスク情報とに基づいてタスクを生成するタスク生成部 1 3 2 と、を備え、タスクが選択されると、タスクのオートコマンドの到達点までの追加パスがロボット 3 の周辺環境情報に基づいて生成され、生成された追加パスとタスクの 2 以上のムーブコマンドとに基づいてロボット 3 が動作する、タスク生成装置を含む。

30

【符号の説明】

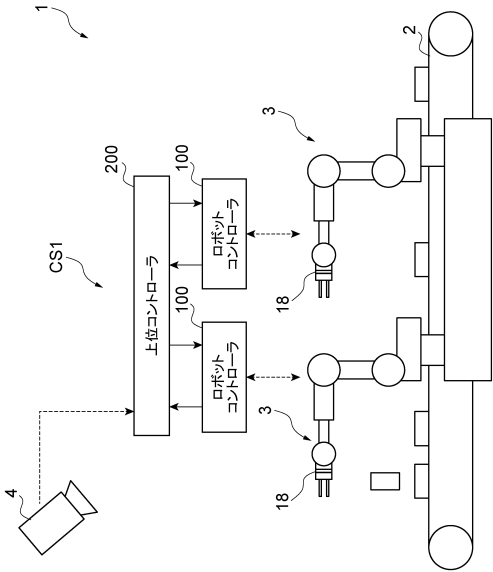
【0 1 4 7】

1 ... ロボットシステム、3 ... ロボット、1 1 1 ... 呼出部、1 1 2 ... パスプラン部、1 1 4 ... 制御部、1 2 1 ... タスク保存部、1 2 3 ... タスク選択部、1 2 4 ... オートコマンド配置部、1 3 1 ... コマンド生成部、1 3 2 ... タスク生成部、1 2 5 ... 環境変化チェック部、2 1 1 ... 情報収集部、3 1 2 ... 干渉チェック部。

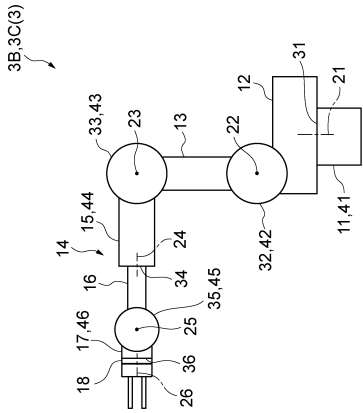
40

【図面】

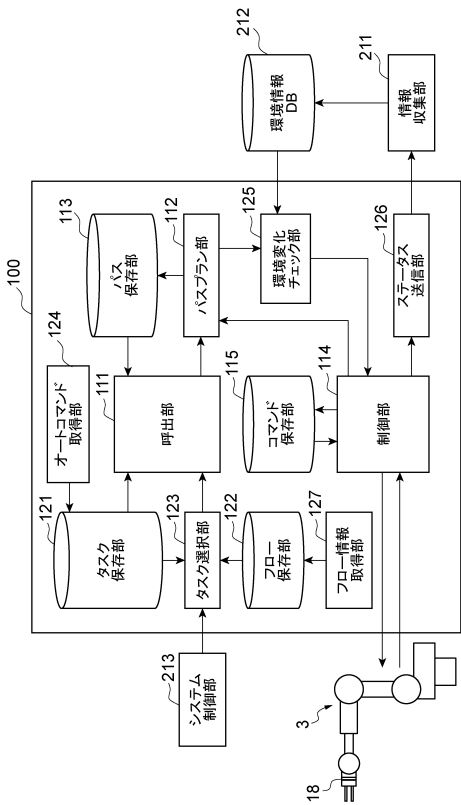
【図 1】



【図 2】



【図 3】



【図 4】

- C1 ~ MoveL (.....)
- C2 ~ MoveS (.....)
- C3 ~ MoveL (.....)
- C4 ~ MoveL (.....)
- C5 ~ MoveAuto (.....)
- C6 ~ MoveL (.....)
- C7 ~ MoveS (.....)
- C8 ~ MoveJ (.....)
- C9 ~ MoveAuto (.....)
- ...

10

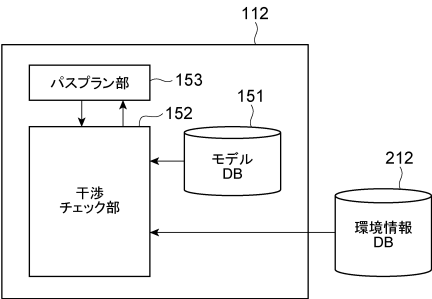
20

30

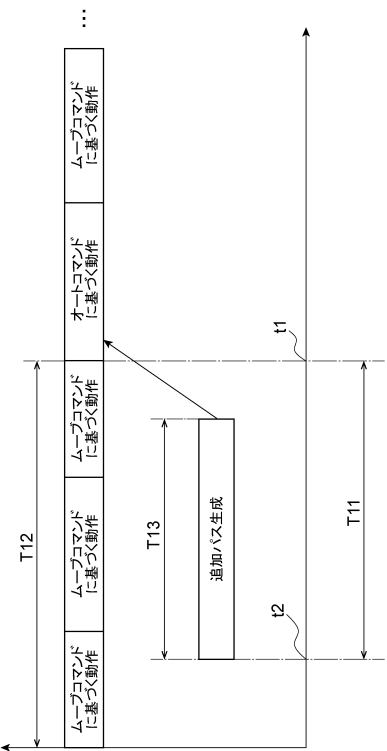
40

50

【図 5】



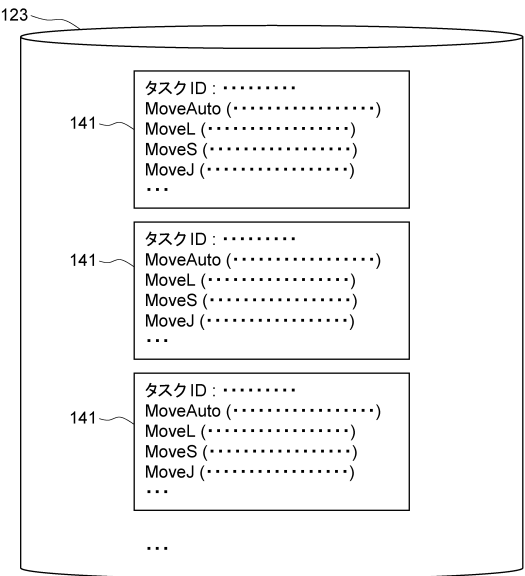
【図 6】



10

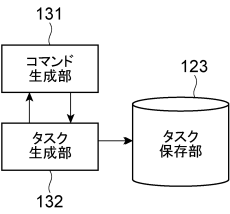
20

【図 7】



30

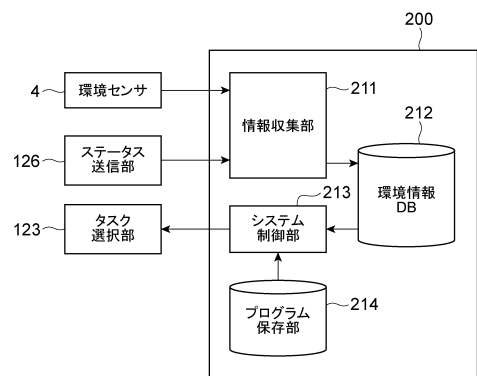
【図 8】



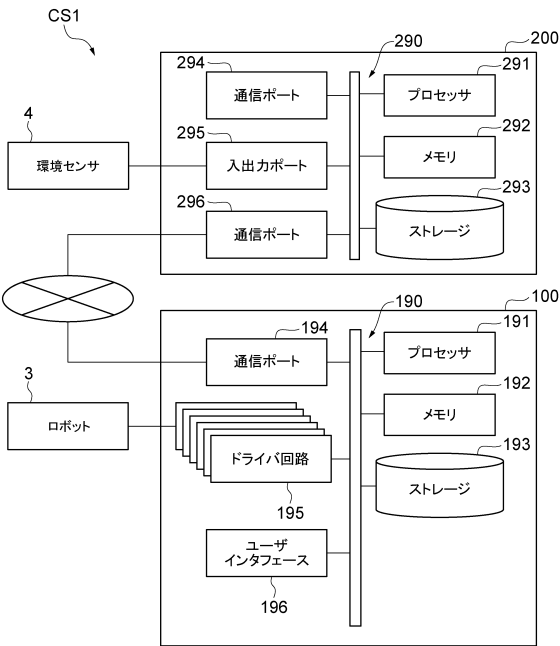
40

50

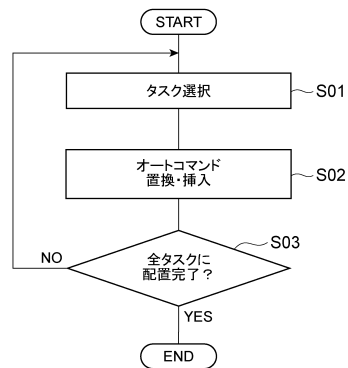
【図 9】



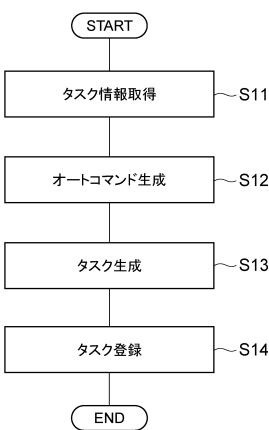
【図 1 0】



【図 1 1】



【図 1 2】



10

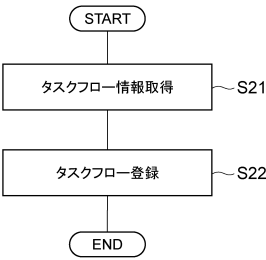
20

30

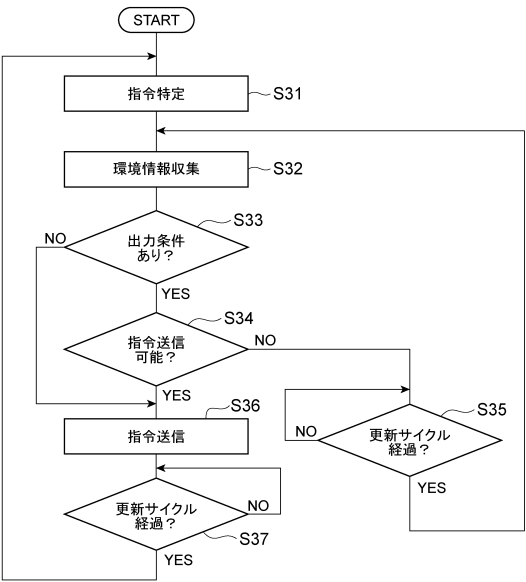
40

50

【図 1 3】



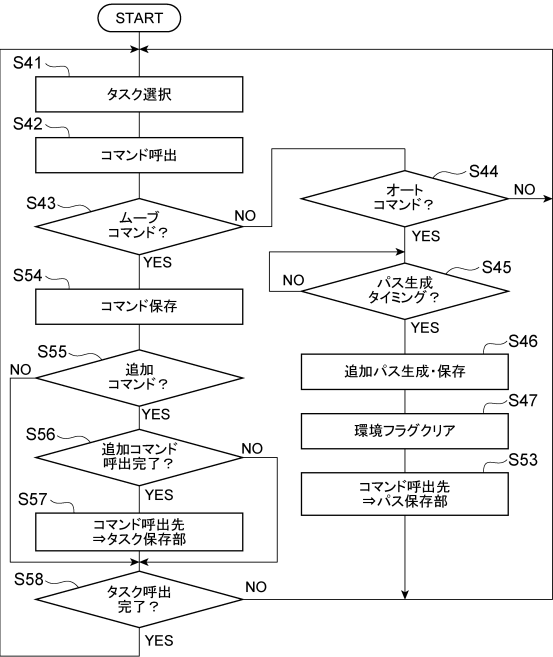
【図 1 4】



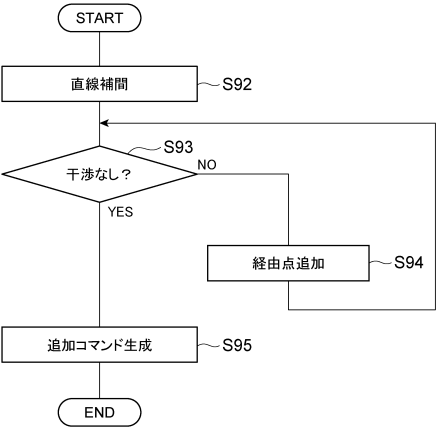
10

20

【図 1 5】



【図 1 6】

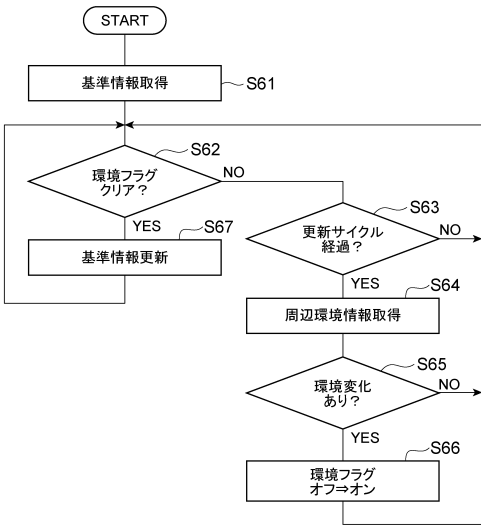


30

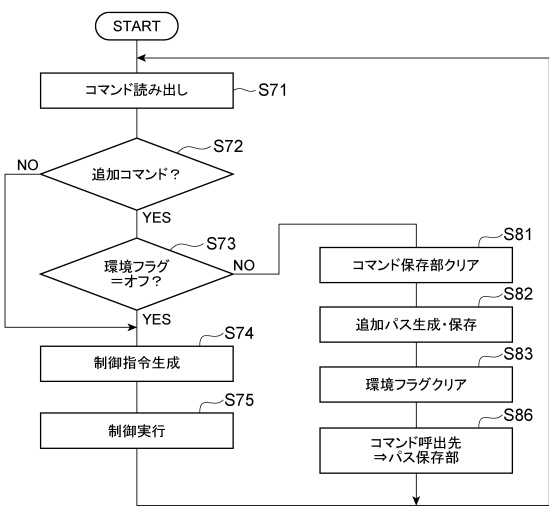
40

50

【図 17】



【図 18】



10

20

30

40

50

フロントページの続き

号 株式会社安川電機内
(72)発明者 有田 裕太
福岡県北九州市八幡西区黒崎城石 2 番 1 号 株式会社安川電機内
審査官 牧 初
(56)参考文献 特開 2 0 1 2 - 1 8 7 6 9 7 (J P , A)
特開昭 6 1 - 2 7 2 8 0 3 (J P , A)
特開 2 0 0 5 - 0 2 5 6 6 8 (J P , A)
特開昭 6 2 - 1 5 7 9 1 1 (J P , A)
特表 2 0 2 2 - 5 0 4 5 9 3 (J P , A)
(58)調査した分野 (Int.Cl. , D B 名)
B 2 5 J 1 / 0 0 - 2 1 / 0 2
G 0 5 B 1 9 / 1 8 - 1 9 / 4 1 6
G 0 5 B 1 9 / 4 2 - 1 9 / 4 6