(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2004-199524
(P2004-199524A)
(43)公開日　平成16年7月15日(2004.7.15)

(51)Int.Cl.[7]	F I		テーマコード（参考）
G05B 19/05	G05B 19/05	B	5H220

審査請求 未請求　請求項の数 6　ＯＬ　（全 24 頁）

(21)出願番号　特願2002-368982(P2002-368982)
(22)出願日　平成14年12月19日(2002.12.19)

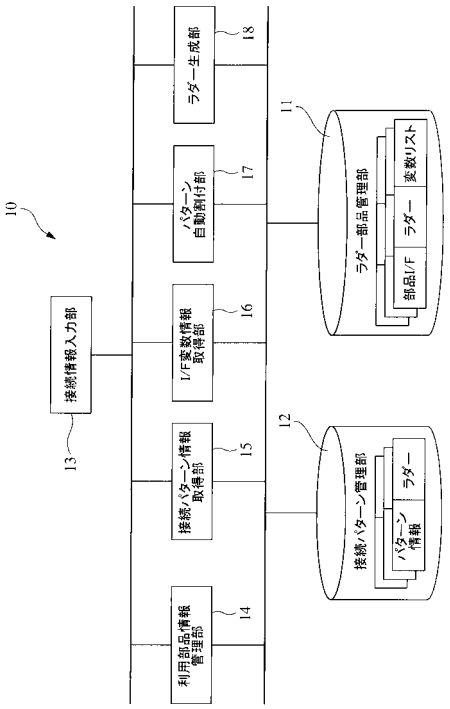
特許法第３０条第１項適用申請有り　平成１４年６月２０日　オムロン株式会社発行の「ＯＭＲＯＮ　ＴＥＣＨＮＩＣＳ　第４２巻　第２号」に発表

(71)出願人　000002945
オムロン株式会社
京都市下京区塩小路通堀川東入南不動堂町８０１番地
(74)代理人　100092598
弁理士　松井　伸一
(72)発明者　山根　仁
京都府京都市下京区塩小路通堀川東入南不動堂町８０１番地　オムロン株式会社内

Ｆターム(参考)　5H220 AA06 BB12 CC05 CX01 DD04 EE08 FF05 GG14 HH01 JJ12 JJ13 JJ24 JJ26 JJ42 JJ53 JJ59

(54)【発明の名称】制御プログラム生成装置及び制御プログラム生成方法並びにプログラム製品

(57)【要約】
【課題】生産設備及びその制御プログラムを適切な単位で部品化することで、効率的な開発が行える制御プログラム作成装置を提供すること
【解決手段】実際の設備単位に部品化されたラダープログラムからなりＩ／Ｆ種別を属性に持つラダー部品と、それを接続する接続パターンを用意し、ラダー部品管理部１１，接続パターン管理部１２に格納する。ラダー部品管理部１１にアクセスし、利用するラダー部品に関する情報を取得し、利用部品情報管理部１４に格納する。接続パターン情報取得部１５は、前記利用するラダー部品の入力Ｉ／Ｆに接続可能な接続パターンを、前記接続パターン管理部から読み出し、ユーザの選択を待って使用する接続パターンを決定する。それら取得された利用されるラダー部品と、それに接続される接続パターンを、ラダー生成部１８が合成することにより、制御プログラムを生成する。
【選択図】　図１１



【特許請求の範囲】
【請求項１】
各入力Ｉ／Ｆ変数に接続ラダーパターン種別を、各出力Ｉ／Ｆ変数に接続変数種別を属性に持つラダー部品を格納するラダー部品管理部と、
前記ラダー部品を接続するための部品接続ロジックをパターン化し、パターン種別を属性として持たせた接続パターンを格納する接続パターン管理部と、
前記ラダー部品管理部に格納されたラダー部品のうち、利用するラダー部品に関する情報を格納する利用部品情報管理部と、
前記利用部品情報管理部に格納された前記利用するラダー部品の入力Ｉ／Ｆに接続可能な接続パターンを、前記属性に基づいて取得する接続パターン情報取得部と、
前記利用部品情報管理部に格納された前記ラダー部品と、それに接続される接続パターンを合成するラダー生成部を持つ制御プログラム生成装置。
【請求項２】
前記接続パターンの各変数と、前記ラダー部品の出力Ｉ／Ｆ変数に種別を設け、
接続パターンに接続可能な前記出力Ｉ／Ｆ変数を自動的に抽出する機能を備えたことを特徴とする請求項１に記載の制御プログラム生成装置。
【請求項３】
ある入力Ｉ／Ｆ変数に対して接続パターンが設定されたとき、他の入力Ｉ／Ｆで同一のＩ／Ｆ種別を持つものに対して同じ接続パターンを自動的に割付けるパターン自動割付部を備えたことを特徴とする請求項１または２に記載の制御プログラム生成装置。
【請求項４】
前記ラダー部品は、実際の設備単位に部品化されたラダープログラムであることを特徴とする請求項１から３の何れか１項に記載の制御プログラム生成装置。
【請求項５】
ラダープログラムで記述された制御プログラムを生成する制御プログラム生成装置におけるプログラム生成方法であって、
Ｉ／Ｆ種別を属性に持つラダー部品を格納するラダー部品管理部から、利用するラダー部品に関する情報を取得するとともに利用部品情報管理部に格納する処理を行い、
前記ラダー部品を接続するための部品接続ロジックをパターン化し、パターン種別を属性として持たせた接続パターンを格納する接続パターン管理部にアクセスし、前記利用部品情報管理部に格納された前記利用するラダー部品に接続可能な接続パターンを取得する処理を行い、
次いで、前記利用部品情報管理部に格納された前記ラダー部品と、それに接続される接続パターンの変数のリンクをとって合成して制御プログラムを生成することを特徴とする制御プログラム生成方法。
【請求項６】
ラダープログラムで記述された制御プログラムを生成するためのプログラム製品であって、
Ｉ／Ｆ種別を属性に持つラダー部品を格納するラダー部品管理部から、利用するラダー部品に関する情報を取得するとともに利用部品情報管理部に格納する処理、
前記利用部品情報管理部に格納された前記利用するラダー部品に接続される接続するパターンを、前記ラダー部品を接続するための部品接続ロジックをパターン化し、パターン種別を属性として持たせた接続パターンを格納する接続パターン管理部から取得する処理、
前記利用部品情報管理部に格納された前記ラダー部品と、それに接続される接続パターンの変数のリンクをとって合成して制御プログラムを生成する処理を含むことを特徴とするプログラム製品。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
この発明は、制御プログラム生成装置及び制御プログラム生成方法並びにプログラム製品

に関するもので、より具体的には、制御システムにおける部品化された制御プログラムを、設計情報を基に自動合成する方式に関する。

【０００２】

【発明の背景】

良く知られているように、各種の生産現場に設置されるファクトリーオートメーション（以下、「ＦＡ」と称する）においては、プログラマブルコントローラ（以下、「ＰＬＣ」と称する）に直接或いはネットワークを介してＩ／Ｏ機器が接続される。ＰＬＣは、係るＩ／Ｏ機器の一種であるスイッチやセンサなどの入力機器からの情報を入力データとして取得し、予め組み込まれたユーザプログラムに従って、係る取得した入力データを用いて演算処理を実行してＩ／Ｏ機器の一種である出力機器への制御内容を決定し、その制御内容に対応する制御データをバルブやアクチュエータ、モータなどの出力機器へ出力することにより、ＦＡシステム全体の制御を行うようになっている。

【０００３】

ＦＡシステムの制御の基幹となるＰＬＣは、入力ユニットに接続された入力機器から入力した信号をＣＰＵユニットのＩ／Ｏメモリに取り込み（ＩＮリフレッシュ）、予め登録されたラダー言語で組まれたユーザプログラムに基づき論理演算をし（演算実行）、その演算実行結果をＩ／Ｏメモリに書き込んで出力ユニットに送り出し（ＯＵＴリフレッシュ）、それにより出力ユニットは出力機器を駆動や停止する制御を行い、その後、通信ネットワークを介して通信処理するようないわゆる周辺処理を行う。このようにＰＬＣは、ＩＮリフレッシュ，演算実行，ＯＵＴリフレッシュ，周辺処理をサイクリックに繰り返し処理する。

【０００４】

そして、上記したラダー言語で組まれたユーザプログラムは、シーケンスプログラムであり、入力機器からの信号に基づく、アクチュエータをＯＮ／ＯＦＦするためのロジックや、動作状態や異常発生中といった監視盤への表示データを生成するためのロジックが記述される。つまり、システムを構成する部品数，制御対象の機器，監視情報などが増えるほど、ラダー言語で記述される接続ロジックも多くなり、ラダー作成工数も多くなり、多大な労力と時間が必要となる。

【０００５】

さらに、ラダー言語は特殊なプログラムであり、専門的な知識が必要となり、習熟度の高いプログラム設計者がプログラミングする必要がある。熟練度によっても、完成までの時間が左右される。

【０００６】

一方、ＦＡシステムの設置現場となる生産工場を立ち上げる場合、その工場に設置，使用する生産ラインの設計→各装置（制御プログラムを含む）の開発→立ち上げを順次実行するようになる。製造業においては、係る生産ラインの設計／開発／立ち上げの期間が長期化すると生産コストが高くなるので、いかに早く立ち上げまで完了するかが大きな課題となっている。そして、ラダー言語によるプログラム開発に要する期間の短縮・労力軽減は、上記した早期立ち上げを図るために重要な要因の１つとなっている。

【０００７】

そこで、効率の良いプログラム作成方法・支援システムの開発が望まれている。また、生産工程の統一化が増えているので、新製品，或いは増産時において生産設備や生産手順が大きく変わることは少ない。従って、現在或いは過去に作成したプログラムの再利用が行えると、作成期間の短縮が期待できる。

【０００８】

現在のプログラム再利用方法として、過去に作成した同種のシステム・機器に対する制御のためのプログラム部分を見つけだすとともに、該当部分をコピー＆ペーストすることで組み合わせていくことがある。しかし、現実には、同種の機器であっても、システムの機器構成によって、インターロックなどの他機器とのやりとりを行うロジックが異なるため、再利用しようとしてコピー＆ペーストしたプログラムを修正しなければならない場合が

ほとんどである。また機器構成が複雑になるとこれらのロジックも複雑になり、修正が困難になるとともに、修正期間も増大する。また、コピー＆ペーストしたプログラムにはＰＬＣ実アドレスも記述されており、再利用する場合には、ラダーに記述されたＰＬＣ実アドレスを全て実際のシステム用のアドレスに手動で変更する必要がある。

【０００９】

この発明は、制御プログラムを制御機器ごとに部品化して再利用することで、システムを構築するイメージでラダープログラムを開発することができ、しかも、ラダー言語に対する習熟した知識が無くても信頼性の高いプログラムを開発することができる制御プログラム生成装置及び制御プログラム生成方法並びにプログラム製品を提供することを目的とする。また、部品間の接続プログラムロジックをパターン化することで部品間接続設計を簡単に行える手法を提供することを目的とする。

【００１０】

【課題を解決するための手段】

この発明による制御プログラム生成方法では、ラダープログラムで記述された制御プログラムを生成する制御プログラム生成装置におけるプログラム生成方法である。

【００１１】

そして、まず、予め操作者（ユーザ）によって登録されたラダー部品及び部品Ｉ／Ｆデータ情報が格納されたラダー部品管理部から、設計中の設備システムで利用するためユーザが指定したラダー部品に関する情報を取得するとともに利用部品情報管理部に格納する処理を行う。また、ラダー部品以外にシステム全体で共通の状態変数などが存在し、ユーザが設定した場合、同時に利用部品情報管理部へ登録する。次いで、部品間のデータ接続を行う。予めユーザによってパターン化された部品間接続ロジック（接続パターン）を格納する接続パターン管理部をアクセスし、前記利用部品情報管理部に格納された前記利用する接続データ入力先ラダー部品の入力Ｉ／Ｆ変数に接続可能な接続パターンを取得する処理を行う。その後、取得した接続パターンの各変数と、接続データ入力基となる前記利用部品情報管理部に格納されたラダー部品の出力Ｉ／Ｆデータのリンクをとって合成することにより制御プログラムを生成するようにした。このとき前記各ラダー部品の入力Ｉ／Ｆ変数に接続可能なパターンを識別するためのパターン種別を属性として持たせ、また、接続パターンにも同様のパターン種別の属性を持たせ、上記接続パターン管理部をアクセスするとき、入力Ｉ／Ｆ変数のパターン属性に一致するラダー部品入力Ｉ／Ｆ変数に接続可能なパターンのみ取得するようにした。このときラダー部品以外のシステム共通の状態変数については、部品入力Ｉ／Ｆ変数と同様に接続パターン設定を行うと同時に、部品出力Ｉ／Ｆ変数のように接続パターンの変数とのリンクを取ることが可能である。

【００１２】

また、この発明による制御プログラム生成装置では、パターン種別を各入力Ｉ／Ｆを属性に持つラダー部品を格納するラダー部品管理部と、前記ラダー部品を接続するための部品接続ロジックをパターン化し、パターン種別を属性として持たせた接続パターンを格納する接続パターン管理部と、前記ラダー部品管理部に格納されたラダー部品のうち、利用するラダー部品に関する情報を格納する利用部品情報管理部と、前記利用部品情報管理部に格納された前記利用するラダー部品の入力Ｉ／Ｆに接続可能な接続パターンを、前記属性に基づいて取得する接続パターン情報取得部と、前記利用部品情報管理部に格納された前記ラダー部品と、それに接続される接続パターンを合成するラダー生成部を備えて構成した。

【００１３】

そして、前記接続パターンの各変数と、前記ラダー部品の出力Ｉ／Ｆ変数に接続変数種別の属性を設け、接続パターンに接続可能な前記出力Ｉ／Ｆ変数を自動的に抽出する機能を備えると良い。さらに、ある入力Ｉ／Ｆ変数に対して接続パターンが設定されたとき、他の入力Ｉ／Ｆで同一のＩ／Ｆ種別を持つものに対して同じ接続パターンを自動的に割付けるパターン自動割付部を備えると良い。但し、これらの機能は必ずしもなくても良い。

【００１４】

上記した各発明によれば、利用するラダー部品を選択すると、そのラダー部品に接続される，或いは接続可能な接続パターンが自動的に取得される。そして、操作者（ユーザ）は、接続パターンが複数取得された場合には、所望の接続パターンを選択することにより特定できるが、この場合でも、多数の接続パターンの中から、接続可能なもののみが抽出されるので、選択対象が限られるので容易に選択できる。そして、結合するラダー部品と接続パターンが決まると、接続パターンの各変数と部品出力Ｉ／Ｆ変数とのリンク設定を行うことにより、制御プログラムが生成される。

【００１５】

よって、ラダー部品や接続パターンにおけるラダープログラム部分は、予め用意しておくことにより、ユーザ（操作者）は、ラダープログラミングを行う必要が無いため、ラダー技術者以外の人でもシステムのラダーを構築することができる。

【００１６】

さらに、システム全体の変数の数が多くなった場合でも、接続可能な変数候補が自動的に選択されるため部品間接続設計工数の削減並びに部品間接続の誤りの減少が可能となる

さらに、Ｉ／Ｆ種別属性によりパターンが自動割付されるため、構成部品が多くなってパターン割付を行うＩ／Ｆの数が多い場合もラダー作成の工数がさほど増加しない。

【００１７】

さらに、前記ラダー部品は、例えばロボットやコンベヤといったシステムに導入される制御機器単位に部品化されたラダープログラムとするとよい。このようにすると、製造設備はメカの作成を担当するメカ設計者とラダープログラムの作成を担当するソフト設計者により作られているので、設備単位に部品化されたラダープログラムを、設備のメカ部品を組み合わせるのと同様に、そのラダープログラム部品を繋ぎ合わせることにより、設備の実稼動プログラムを生成することができる。これにより、ラダープログラミングにおける習熟度を排除し、上流設計工程において実稼動プログラムの作成ができる。

【００１８】

また、本発明に係るプログラム製品は、ラダープログラムで記述された制御プログラムを生成するためのプログラム製品であって、Ｉ／Ｆ種別を属性に持つラダー部品を格納するラダー部品管理部から、利用するラダー部品に関する情報を取得するとともに利用部品情報管理部に格納する処理と、前記利用部品情報管理部に格納された前記利用するラダー部品に接続される接続するパターンを、前記ラダー部品を接続するための部品接続ロジックをパターン化し、パターン種別を属性として持たせた接続パターンを格納する接続パターン管理部から取得する処理と、前記利用部品情報管理部に格納された前記ラダー部品と、それに接続される接続パターンの変数のリンクをとって合成して制御プログラムを生成する処理を含むようにした。

【００１９】

【発明の実施の形態】

図１１は、本発明の好適な一実施の形態を示している。まず本実施の形態では、本システムを利用する前に、ラダープログラムを制御機器単位に部品化してラダー部品（ラダープログラム部品）を作成しラダー部品管理部１１に登録しておく必要がある。また、このときラダープログラム部品とともに、ラダープログラムで使用する変数のリストと部品へのデータ入力，出力，入出力などの部品Ｉ／Ｆを登録する。そして、このラダー部品は、少なくとも制御機器の種類に相当するものを用意する。さらに、ラダー部品同士を接続するための接続パターン、つまり、部品間接続ロジックをパターン化したプログラム部品をラダー言語を用いて作成したものを用意し接続パターン管理部１２へ登録しておく。そして、設備システムのラダープログラム作成時には、実際の設備のメカ部品を組み合わせるのと同様に、使用するラダー部品を組み合わせ、部品間を接続パターンを用いて繋ぎ合わせることにより、設備システム全体の制御プログラムを生成するようにしている。

【００２０】

そして、本実施の形態で用いるラダー部品は、制御機器と１：１に対応したロジックからなり、部品へのデータ入力，出力，入出力などのＩ／Ｆ変数が定義されたものであり、実

際に制御機器とやりとりする信号データの変数は部品内部に隠蔽されている。なお、これらＩ／Ｆ変数において、入力Ｉ／Ｆ変数に対して接続可能な接続パターンを識別可能にするための接続パターン種別を属性として持たせることにより、接続パターンと結合する際には、接続パターン種別が、入力Ｉ／Ｆの属性と接続パターンの属性とで一致することを条件として結合可能としている。なお、このラダー部品並びに接続パターンの詳細な構成は、具体例をあげつつ後述する。

【００２１】

上記したラダー部品と接続パターンを適宜用いて最終的なラダー言語で記述された制御プログラムを生成するためのツール１０としては、以下のような構成を採っている。すなわち、まずデータベースとして、ラダー部品を格納するラダー部品管理部１１と、接続パターンを格納する接続パターン管理部１２を備えている。

【００２２】

そして、これら各データベースに格納された各種情報を読み出して、プログラムを作成するツールの本体構成は、ラダー部品の選択や各ラダー部品が持つ部品Ｉ／Ｆへの接続パターン割付及び割付パターンと他のラダー部品との接続のための設定を行う接続情報入力部１３と、利用するラダー部品のリスト情報を管理する利用部品情報管理部１４と、部品入力Ｉ／Ｆ変数への接続可能パターン候補としてＩ／Ｆ変数に定義された接続パターン種別と同一種別の接続パターンを接続パターン管理部１２から取得する接続パターン情報取得部１５と、部品の入力Ｉ／Ｆリストと各Ｉ／Ｆの接続パターン種別の取得及びパターン変数へのリンク可能な変数候補としてパターン変数に定義された接続変数種別と同一種別の他部品の出力Ｉ／Ｆリストをラダー部品管理部から取得するＩ／Ｆ変数情報取得部１６と、同一Ｉ／Ｆ種別を持つ全ての部品Ｉ／Ｆに同一パターンを割り付けるパターン自動割付部１７と、設計情報とラダー部品，接続パターンを基にラダー言語で記述される制御プログラム（システムラダー）を構築するとともに物理アドレスの自動割付を行うラダー生成部１８を備えている。整理すると、次の表のようになる。

【００２３】

【表１】

ラダー部品管理部	次の属性を持つ３つの要素を関連付けたものを１部品とし、全部品を管理する。 ・部品 I/F 情報：部品名、部品 I/F 情報（入力、出力 I/F となる変数及び I/F 変数属性（入力 I/F：接続パターン種別、出力 I/F：接続変数種別）） ・ラダー：部品化されたラダープログラム ・変数リスト：部品ラダーで用いられている変数のリスト。論理名、データ型などを格納。
接続パターン管理部	部品間の接続ロジックをパターン化したものを管理する。 次の属性を持つ２つの要素を関連付けたものを１パターンとし、全パターンを管理する。 ・パターン情報：パターン名、パターン属性 ・ラダー：パターンロジックをラダーで記述したもの
接続パターン情報取得部	接続パターン管理部より接続パターン属性が一致するパターンの検索、及びパターン情報（データ型、パターンラダー）を取得する。
I/F 変数情報取得部	各部品の I/F 変数リストの取得、及び接続変数属性が一致する出力 I/F 変数リストを取得する。
パターン自動割付部	登録されている部品で同じ I/F 種別をもつ I/F に対して同じパターンを割り付ける。
ラダー生成部	ラダー部品とパターンからシステムのラダーを構築する。
物理アドレス割付部	ラダー、パターンの I/O の論理変数に物理アドレスを割り付ける。
マウス入力部	パターン選択などのマウスからの入力を受け付ける。
接続情報入力部	部品 I/F 情報の画面表示、

【００２４】
ラダー部品管理部１１は、ラダーと、変数リストと、部品Ｉ／Ｆ情報の３つの要素を関連付けたものを１つの部品とし、全部品を管理、つまり格納・保管するする。部品名は、部品を識別するための任意の名称であり、例えば、「コンベヤ」、「ピッキングロボット」、「ペイントロボット」のように実際の制御機器名に対応する名称を付すと、わかりやすいので好ましい。もちろん、それ以外の名称を付すのも自由である。
【００２５】
ラダーは、部品化されたラダープログラムを格納するものである。また、変数リストは部品化されたラダープログラムで用いられている変数のリストであり、論理名、データ型などを格納する。
【００２６】
また、部品Ｉ／Ｆ情報は部品の入力Ｉ／Ｆ、出力Ｉ／Ｆ、入出力Ｉ／Ｆの情報を格納するものである。入力Ｉ／Ｆ情報は、変数リストのうち部品の入力Ｉ／Ｆとなる変数のリストであり、入力Ｉ／Ｆ変数の名称である「入力Ｉ／Ｆ変数名」と、パターン情報のパターン型に対応付けられる接続可能なパターンの種別である「接続パターン種別」の２項目を“，”で区切り記述する。また、出力部品Ｉ／Ｆ情報は、変数リストのうち部品の出力Ｉ／Ｆとなる変数のリストであり、出力Ｉ／Ｆ変数の名称である「出力Ｉ／Ｆ変数名」と、接続パターンラダーにおける各変数とリンク設定可能とするための種別である「接続変数種別」の２項目を“，”で区切り記述する。具体的には、図１９に示すようなデータ構造となる。部品名称と、部品を識別するための任意の名称を書くと、下記表２のようになる。
【００２７】
【表２】

部品名称		部品を識別するための任意の名称
入力I/F情報		部品の入力I/F変数のリスト。下の2項目を“,”で区切り記述する。
	入力I/F変数名	入力I/F変数の名称
	接続パターン種別	接続可能なパターンの種別。パターン情報のPatternTypeに対応付けられる。
出力I/F情報		部品の出力I/F変数のリスト。下の2項目を“,”で区切り記述する。
	出力I/F変数名	出力I/F変数の名称
	接続変数種別	リンク可能なパターン変数の種別。パターン情報のPatternProgramで記述されたパターン変数の種別に対応付けられる。

【００２８】
接続パターン管理部１２は、部品間の接続ロジックをパターン化したものを管理、つまり格納・保管する。つまり、パターン情報とラダーの２つの要素を関連付けたものを１パターンとし、全パターンを管理する。ここで、パターン情報とは、パターン名とパターン属性を含む。より具体的には、パターンを識別するための任意の名称であるパターン名と、部品入力Ｉ／Ｆの「接続パターン種別」と対応した接続可能な部品入力Ｉ／Ｆを識別するための「パターン種別」と、パターンが接続可能とする入力Ｉ／Ｆのデータ型を有する。ラダーは、パターンロジックをラダーで記述したものである。このとき、パターン内変数の接続変数種別は、プログラム内で記述する。そして、具体的な一例を示すと、図１８のようになる。そして、ここでの項目は表３に説明する。
【００２９】
【表３】

フィールド名	説　明
PatternName	パターン名。
PatternType	パターン種別
DataType	パターンが接続可能とする入力I/Fのデータ型。
PatternProgram	パターンのラダープログラム。 パターン内変数の接続変数種別はプログラム内で記述する。

【００３０】
次に、図１のような設備システムの制御ラダープログラムを、本システムを用いて作成する際の実際の動作を説明する。図１はワークを塗装する設備（ライン）の構成を表したものであり、各設備名は、表４のようになっていて、設備全体の動作は図２に表したフローチャートのようになっている。なお、この図１のような設備システムにとって、従来作成されていたラダープログラム（部品化前のラダープログラム）は、図３のようになっていた。この図３を見るとわかるように、ラダープログラムは設備ごとに制御ロジックをまとめて記述する。また、同一設備機種においては、出力制御データ項目は同じである。しかし、同一設備機種においても、起動要件などの制御ロジックがデバイスごとに異なる事情はある（これは、ラダーの切り貼りだけの再利用は不可能であることを意味している）。そして、ＰＬＣアドレスがラダープログラムに直接記述されている（これは、もし再利用するとすれば、その都度ＰＬＣアドレスを割り付け直さなければならないということを意味する）。
【００３１】
【表４】

設備名	動作説明	PLCへの入力信号	PLCへの出力信号
ワーク投入ロボット	作動開始信号受信で塗装前ワーク置へ移動→ワークチャック→コンベヤへワーク移動→ワークリリースの作業を行い作業完了信号を送信する。 原姿勢戻信号受信で基本姿勢へ戻る。	ロボット原位置 作業完了 異常発生	ロボット作動 ロボット作動停止 ロボット原位置戻
ワーク排出ロボット	作動開始信号受信でコンベヤへ移動→ワークチャック→塗装後ワーク置へワーク移動→ワークリリースの作業を行い作業完了信号を送信する。 原姿勢戻信号受信で基本姿勢へ戻る。	同上	同上
二色ペイントロボット	作業開始信号受信でワークの塗装を行う。ワークタイプによって塗装色を青と赤に切り替える。塗装完了時に作業完了信号を送信する。	同上	同上
メインコンベヤ	ワークを搬送する。 搬送開始信号、搬送停止信号で作動／停止を行う。 投入位置検出センサ、ワーク機種読み取りセンサ、塗装位置検出センサ、搬出位置検出センサを含む。	投入位置検出センサ 塗装位置検出センサ 搬出位置検出センサ ワーク機種読み取りデータ異常発生	コンベヤ作動 コンベヤ停止
操作盤	操作者がライン操作を行う。 運転ON、緊急停止のスイッチを持つ。	運転ON 緊急停止	異常発生中

【００３２】
まず、ワーク投入ロボット２１でワーク２２がメインコンベヤ２３に投入される。ワーク投入ロボット２１から作業完了信号を受け取ること、または投入位置検出センサ２８の検出領域にワーク２２をセンシングすることでワーク２２が投入されたことを検知し、メインコンベヤ２３が搬送を開始する。次に、塗装位置検出センサ２４によりワーク２２が塗装位置まで到着したことを検知し、メインコンベヤ２３が停止するとともに二色ペイントロボット２５が塗装を開始する。塗装が完了すると、再度メインコンベヤ２３が搬送を開始する。次に、排出位置検出センサ２６によりワーク２２が排出位置まで到着したことを検知し、メインコンベヤ２３が停止するとともにワーク排出ロボット２７が排出作業を開始する。ワーク排出ロボット２７が作業を完了するとワーク２２は図示されているように所定位置に置かれた状態になる。以上の作業を、運転開始時に設定されたワーク数分の塗装が完了するまで繰り返す。なお、図中符号２９は、メインコンベヤ２３上を搬送されるワーク２２の機種を検知するワーク機種読み取りセンサであり、符号３０は操作盤である。

【００３３】
このようなシステムの制御プログラムを作成するにあたり、ラダー部品管理部１１へ、図

４で示すような「ピッキングロボット」と、図５で示すような「塗装ロボット」部品と、図６で示すような「コンベヤ」部品を予め登録しておく必要がある。図４のピッキングロボット向けに制御されるラダー部品としてプログラミングされている中身は、作動フラグを受信しワーク移動を開始すること，移動完了時に作業完了信号をＰＬＣへ送信すること，原位置戻フラグを受信しロボット原姿勢へ戻ること，作業停止信号を受信した場合にロボット作動を停止すること，ロボット異常発生時に異常発生信号をＰＬＣへ送信することなどである。なお、その詳細は下記の表５の通りである。

【００３４】

【表５】

入出力種別	変数名	接続パターン種別	接続変数種別	説　明
部品入力	作動条件	作動入力	—	ロボット作動条件フラグ。
	原位置戻条件	作動入力	—	ロボット原位置戻条件フラグ。
	作業停止条件	停止入力	—	作動中ロボットの強制停止フラグ。
部品出力	作動	—	作動指令	ロボットへの作動起動フラグ。作動中の状態フラグとしても利用する。
	原位置戻	—	作動指令	ロボットへの原位置戻起動フラグ。原位置戻中の状態フラグとしても利用する。
	作業停止	—	作動指令	ロボットへの強制作業停止フラグ。
	作業完了	—	運転状態	ロボットからの作業完了信号。
	原位置	—	運転状態	ロボットからの原位置停止状態信号。
	異常発生	—	異常データ	ロボットからの異常発生フラグ。
内部変数	—			

【００３５】

また、図５で示したのは、「塗装ロボット」で制御されるラダー部品である。これは、ワークを塗装するためのロボット部品であり、そこでプログラミングされた内容は、作動フラグを受信し塗装を開始すること，塗装完了時（ロボット原位置戻完了時）に作業完了信号をＰＬＣへ送信すること，作業停止信号を受信した場合にロボット作動を停止すること，ロボット異常発生時は異常発生信号をＰＬＣへ送信することなどである。部品入力と部品出力の詳細は表６の通りである。

【００３６】

【表６】

入出力種別	変数名	接続パターン種別	接続変数種別	説　明
部品入力	作動条件	作動入力	—	ロボット作動条件フラグ。
	作業停止条件	停止入力	—	作動中ロボットの強制停止フラグ。
	塗装パターンデータ	プロパティ設定	—	ロボットへの塗装パターンデータ。
部品出力	作動	—	作動指令	ロボットへの作動起動フラグ。作動中の状態フラグとしても利用する。
	作業停止	—	作動指令	ロボットへの強制作業停止フラグ。
	作業完了	—	運転状態	ロボットからの作業完了信号。
	原位置	—	運転状態	ロボットからの原位置停止状態信号。
	異常発生	—	異常データ	ロボットからの異常発生フラグ。
内部変数	—			

【００３７】
また、図６で示したのは「コンベヤ」を制御するためのラダー部品である。これは、ワークを搬送するための「コンベヤ部品」であり、そこでプログラミングされている内容は、コンベヤ作動フラグを受信しワーク搬送を開始すること、ロボット位置検出センサまたは搬出位置検出センサ検知時に搬送を停止すること、コンベヤ異常発生時に異常発生信号をＰＬＣへ送信することなどである。同様に表にまとめると次のようになる。
【００３８】
【表７】

入出力種別	変数名	接続パターン種別	接続変数種別	説　明
部品入力	コンベヤ作動条件	作動入力	—	コンベヤ搬送作動条件フラグ。
	コンベヤ強制停止条件	停止入力	—	搬送中コンベヤの強制停止フラグ。
部品出力	コンベヤ作動	—	作動指令	コンベヤへの搬送起動フラグ。作動中の状態フラグとしても利用する。
	コンベヤ停止	—	作動指令	コンベヤへの強制搬送停止フラグ。
	ワーク機種読み取りデータ	—	ワークデータ	ワーク機種読み取りセンサで読み取ったワーク機種データ。
	投入位置検出センサ	—	ワーク検出	ワーク投入位置に配置されたセンサのワーク検出信号。
	ロボット位置検出センサ	—	ワーク検出	ロボット作業位置に配置されたセンサのワーク検出信号。
	排出位置検出センサ	—	ワーク検出	ワーク排出位置に配置されたセンサのワーク検出信号。
	異常発生	—	異常データ	コンベヤからの異常発生フラグ。
内部変数	—			

【００３９】
次に、実際の動作を説明しながら、各部の詳細な機能の説明をする。図１４は、ラダー部品並びに接続パターンを選択し、接続する処理機能を説明するフローチャートである。
【００４０】
図１４に示すように、まずユーザから入力された利用部品登録要求を受け付ける（ＳＴ１）。この要求は、図示省略する入力装置から与えられるもので、使用する部品名等により特定される。つまり、ツールは、パソコンに所定のプログラムをインストールすることにより生成されるので、そのパソコンに用意されたキーボードやポインティングデバイスその他の入力装置を用い、ユーザが係る利用部品の部品名を登録する。このとき、ラダー部品を設備単位に部品化するとともに、その部品名を実際の装置の名称に対応するように設定しておくと、実際のシステムを構成する制御機器が決定すると、その装置の名称を登録することにより、ラダー言語に対する習熟した知識がない人であっても、利用部品の登録が簡単に行える。よって、製造設備におけるメカの作成を担当するメカ設計者が、この利用部品の登録をすることも可能となる。
【００４１】
なおここで、本設備システムを構築するにあたり、ラダー部品管理部１１に登録されているラダー部品のうち、「ピッキングロボット」部品を２つ、「塗装ロボット」部品を１つ、「コンベヤ」部品を１つ、インスタンスとして利用するようにユーザから要求を受けるものとする。ここでインスタンス名を「ピッキングロボット」部品に対してそれぞれ「ワーク投入ロボット」、「ワーク排出ロボット」とし、「塗装ロボット」部品に対して「二色ペイントロボット」とし、「コンベヤ」部品に対して「メインコンベヤ」とする。さらに、システム共通の状態変数として塗装すべき残りのワーク数を示す「ワーク残」と、シ

ステム全体における異常発生状態を示す「異常発生中」の２つをユーザから登録要求があるものとする。

【００４２】

そして、このようにユーザから入力された利用部品登録要求は、利用部品情報管理部１４に渡される。利用部品情報管理部１４は、受け取った部品名にしたがってラダー部品管理部１１にアクセスし、該当するラダー部品を取得し、利用部品情報管理部１４に格納する（ＳＴ２，ＳＴ３）。これにより、利用部品情報管理部１４には、制御プログラムの作成対象となるシステムを構成する機器に対応するラダー部品（部品Ｉ／Ｆ情報，ラダー，変数リスト）が登録される。なお、この説明では、利用部品情報管理部１４が必要なラダー部品を読み出す（取得する）機能と、その取得したラダー部品の情報を記憶保持する機能を持たせたが、取得する機能は別途用意しても良い。

【００４３】

上記ステップ１から３の処理は、接続表示要求があるまで行われる（ＳＴ４）。すなわち、ユーザは、利用部品登録要求を順次行い、利用する全ての部品登録が完了するまで上記のステップ１～３が繰り返される。そして、全ての登録要求が済むと、接続情報入力部１３を介して接続情報表示要求を入力するようにしている。

【００４４】

この接続情報表示要求に従い、登録部品，部品Ｉ／Ｆ情報を取得するとともに画面に表示する（ＳＴ５）。具体的には、接続情報入力部１３は、登録部品つまり利用部品情報管理部１４に格納されているラダー部品（部品Ｉ／Ｆ情報含む）を取得するとともに、ツール（パソコン）の表示装置の画面に部品Ｉ／Ｆリストを表示する。

【００４５】

この部品Ｉ／Ｆリストは、例えば図２０に示すような表形式で表示することができる。もちろん、表示形式はこれに限ることはない。そして、ここでの表示例は、「登録部品名」，「部品Ｉ／Ｆ入力変数」，「データ型」，「Ｉ／Ｆ接続パターン」，「パターン接続元変数」並びに「パターン接続先変数」を関連づけたデータ構造となっている。そして、「登録部品名」は、「利用するために登録した部品のインスタンス名」であり、「部品Ｉ／Ｆ入力変数」は、「各部品の入力Ｉ／Ｆのリスト」であり、「データ型」は、「入力Ｉ／Ｆのデータ型（ＢＯＯＬ、ＣＨＡＮＮＥＬなど）」である。そして、ステップ５の実行により表示されるのは、この「登録部品名」，「部品Ｉ／Ｆ入力変数」並びに「データ型」の３種類のデータであり、それ以外の項目は実際には空欄となる。例えば、「ワーク投入ロボット」は入力Ｉ／Ｆ変数として「原位置戻条件」，「作業停止条件」，「作動条件」をもつことを示している。なお、「Ｉ／Ｆ接続パターン」は、入力Ｉ／Ｆに割り付けられた接続パターン名であり、割付完了後に表示内容が更新される。同様に、「パターン接続元変数」は、接続パターンラダー内の変数名であり、「パターン接続先変数」は、パターン接続元変数名とデータがリンクするよう設定された他部品の出力Ｉ／Ｆ変数名である。これらの項目名と具体的な説明の関係を表にまとめると下記表８のようになる。

【００４６】

【表８】

項　目	説　明
登録部品名	利用するために登録した部品名
部品 I/F 入力変数	各部品の入力 I/F のリスト
データ型	入力 I/F のデータ型。(BOOL、CHANNEL など)
I/F 接続パターン	入力 I/F に割り付けられた接続パターン名。割付完了後表示が更新される
パターン接続元変数	接続パターンラダー内の入力変数名
パターン接続先変数	パターン接続元変数名に設定された他部品の出力 I/F 変数名

【００４７】
次いで、ユーザからパターン設定要求が入力されるのを待つ（ＳＴ６）。このパターン設定要求は、例えば、図２０に示した部品Ｉ／Ｆリストのうちの「部品Ｉ／Ｆ入力変数」の処理対象の欄をクリックすることにより行われる。
【００４８】
そして、係る要求を受けると（ステップ６でＹｅｓ）、部品入力Ｉ／Ｆへのパターン割付処理が実行され（ＳＴ７）、指定されたラダー部品同士の結合処理、つまり、所望の接続パターンを決定するとともに、その接続パターンを用いてラダー部品同士を連結する処理を実行する（詳細は後述する）。
【００４９】
そして、ステップ７を実行して決定された情報を、部品入力Ｉ／Ｆリストに反映して、処理を終了する（ＳＴ８）。つまり、図２０に示すように、部品入力リスト画面に設定されたパターン情報を表示する。これにより、ステップ５の状態では空欄であった「Ｉ／Ｆ接続パターン」以降の欄が図２０に示すように設定された状態となる。例えば、「ワーク投入ロボット」の入力Ｉ／Ｆ「原位置戻条件」に対して接続パターン「アップタイミング」が割り付けられており、パターン変数「取得基信号」と「ワーク投入ロボット」インスタンスの出力Ｉ／Ｆ変数「作業完了」がリンク設定されていることを示している。
【００５０】
一方、上記したステップ７の部品入力Ｉ／Ｆへのパターン割付処理は、接続情報入力部１３，接続パターン情報取得部１５，Ｉ／Ｆ変数情報取得部１６，パターン自動割付部１７が適宜動作して実行されるもので、具体的には図１５に示すフローチャートを実施するようになる。
【００５１】
まず、接続パターン情報取得部１５が、接続パターン管理部１２にアクセスし、処理対象のラダー部品に接続可能な接続パターンを取得する（ＳＴ１１）。つまり、処理対象のラダー部品の入力Ｉ／Ｆ変数の接続パターン属性と同じ属性を持つ接続パターンを検出する。そして、該当する接続パターンが複数存在する場合には、その複数の接続パターンを全て取得する。そして、モニタ画面に全て表示する。このときの表示画面の一例としては、例えば図２１に示す接続パターン設定画面とすることができる。
【００５２】
この接続パターン設定画面は、「部品名」，「Ｉ／Ｆ名」，「データ型」，「パターン」，「コメント」，「プログラム」並びに「変数接続定義」の欄が用意されている。これらの各国項目の具体的な内容は、下記表９の通りである。
【００５３】
【表９】

項　目	説　明
部品名	パターンを割り付ける I/F の部品名
I/F 名	パターンを割り付ける I/F 名
データ型	I/F のデータ型
パターン	割付可能なパターンをリストから選択する
コメント	選択されたパターンのコメント。(表示のみ)
プログラム	選択されたパターンのラダープログラム。(編集ボタンをクリックすることで編集可能)
変数接続定義	パターンラダーの入力変数のリスト、及び他部品の出力 I/F 割付状況の表示。変数割付ボタンをクリックすることで出力 I/F 割付を行う。

【００５４】
すなわち、各欄に記述される内容は、「部品名」は、パターンを割り付けるＩ／Ｆの部品名が表示される。また、「Ｉ／Ｆ名」には、パターンを割り付けるＩ／Ｆ名が表示される。「データ型」は、Ｉ／Ｆのデータ型が表示される。「パターン」の欄は、割付可能なパターンが表示される領域であり、プルダウンメニュー方式で割り付け可能なパターンリストが表示され、そのリストから所望のパターンをユーザが選択する（図示の場合は、「作業停止条件」が選択されている）。
【００５５】
さらに、「コメント」は、上記プルダウンメニューの中から選択されたパターンについて付されているコメントが表示される。さらに、「プログラム」の欄には、上記選択されたパターンのラダープログラムが表示される。このラダープログラムは、その欄外に用意された「編集ボタン」をクリックすることで編集可能となる。これにより、類似するラダープログラムを持つパターンを呼び出し、一部を修正することにより、より適した接続パターンを作成することができる。さらに、「変数接続定義」は、パターンラダーにおける変数のリスト各変数と他部品の出力Ｉ／Ｆとのリンク設定状況を表示する領域である。そして、その欄外の「変数割付ボタン」をクリックすることで他部品の出力Ｉ／Ｆ変数とのリンク設定を行うようになる。
【００５６】
すなわち、プルダウンメニューから操作者による割付けパターンの選択が行われると（ＳＴ１２）、Ｉ／Ｆ変数情報取得部１６が、パターン内変数の接続変数属性と一致する他部品の出力Ｉ／Ｆリストをラダー部品管理部の部品Ｉ／Ｆ情報から取得し、画面へリスト表示する（ＳＴ１３）。
【００５７】
すると、表示されたリストに対し、操作者が関連付けるパターン変数と他部品出力Ｉ／Ｆ変数を指定するので、係る指定に伴い、それらに対するリンクを設定する（ＳＴ１４）。そして、係るリンク設定を全てのパターン変数に対して行うまで、ステップ１３，１４の処理を繰り返し実行する（ＳＴ１５）。
【００５８】
このようにして、全てのパターン内変数に対してリンク設定が行われるとパターン設定完了となる（ＳＴ１６）。さらに、全部品の入力Ｉ／Ｆ変数に対して、パターンを割付けた入力Ｉ／Ｆ変数と同じ接続パターン種別の属性を持つ変数があった場合、設定されたパターンをその変数にも設定する（ＳＴ１７，ＳＴ１８）。この自動的に行う設定は、登録されている部品で同じＩ／Ｆ種別をもつＩ／Ｆに対して同じパターンを割り付ける機能を持つパターン自動割付部１７が行う。
【００５９】

例えば「ワーク投入ロボット」の「作業停止条件」と「ワーク排出ロボット」の「作業停止条件」と「メインコンベヤ」の「コンベヤ停止条件」は、同じ接続パターン種別「停止条件」を持つため、「ワーク投入ロボット」の「作業停止条件」に対して接続パターンが設定された時点で、同じ接続パターンが前記他部品の入力Ｉ／Ｆ変数に対しても割り付ける。

【００６０】

上記した接続パターンの選択並びに入出力Ｉ／Ｆのリンクの設定処理は、実際には例えば図１３，図１２に示すようになる。なお、図１３は、図１２の一部を拡大して示している。そして、利用部品情報管理部１４には、ラダー部品「ピッキングロボット」が２つ，「塗装ロボット」，「コンベヤ」が１つずつ登録されている。インスタンス名はそれぞれ「ワーク投入ロボット」，「ワーク排出ロボット」，「塗装ロボット」，「メインコンベヤ」となっている。そして、部品「ワーク投入ロボット」は、入力Ｉ／Ｆとして「作動条件」（接続パターン種別：作動条件），「原位置戻条件」（作動条件），「作業停止条件」（停止条件）を持ち、各Ｉ／Ｆに対してパターンを割り付ける。そして、具体的な割付処理は、いかのとおりである。

【００６１】

（１）まず、入力Ｉ／Ｆ変数「作動条件」の接続パターン種別「作動条件」の属性と同じパターン種別を持つ接続パターンを、接続パターン管理部１２から取得し、ユーザが「ロボット作動（ワーク残チェック有）」の接続パターンを選択する（図１３中▲１▼）。

【００６２】

（２）接続パターン「ロボット作動（ワーク残チェック有）」内の変数「排出ロボット停止中」の接続変数種別「運転状態」と同じ接続変数種別を持つほか部品の出力Ｉ／Ｆ変数を表示し、その中から部品「ワーク排出ロボット」の出力Ｉ／Ｆ変数「作業完了」を選択し、リンク設定を行う（図１３中▲２▼）。

【００６３】

（３）上記した（２）の手順でパターン内の全ての変数に対してリンク設定を行い、接続パターンと他部品間のリンク設定を完了する（図１３中▲３▼）。

【００６４】

（４）上記した（１）～（３）の手順で部品「ワーク投入ロボット」の各入力Ｉ／Ｆに対して接続パターン設定を行い、「ワーク投入ロボット」に対する接続パターン設定を完了する（図１３中▲４▼）。このとき「ワーク投入ロボット」の入力Ｉ／Ｆ「作業停止条件」と、「ワーク排出ロボット」の入力Ｉ／Ｆ「作業停止条件」と、「二色ペイントロボット」の入力Ｉ／Ｆ「作業停止条件」と、「メインコンベヤ」の入力Ｉ／Ｆ「コンベヤ停止条件」の接続パターン種別が、「停止入力」で同じであり、「ワーク投入ロボット」の入力Ｉ／Ｆ「作業停止条件」への接続パターン設定が完了したときに、パターン自動割付部１７がこれら入力Ｉ／Ｆに対しても同じ接続パターンを設定する。

【００６５】

（５）上記した（１）～（４）の手順で他部品「ワーク排出ロボット」、「二色ペイントロボット」、「メインコンベヤ」の入力Ｉ／Ｆ、及びシステム共通の状態変数に対してパターン設定を行い、全設定を完了する。このとき（４）で同じ接続パターン種別で自動的に接続パターンが割り付けられた入力Ｉ／Ｆに対して、接続パターンの変更やリンク内容の変更を行うことも可能である。

【００６６】

なお、実際の接続パターンの例を図７から図１０に示す。図７は、ワーク投入ロボット用作動入力用パターンであり、図１３の中央上の位置にある接続パターンである。これは、関連する設備が停止中で加工ワークに残りがある場合のみ作動するロジックである。図７中のパターン変数名、接続変数種別、変数の説明は、表１０のとおりである。

【００６７】

【表１０】

パターン変数名	接続変数種別	変数説明
電源ON	システムデータ	システムの電源がON状態のときONとなる。
排出ロボット停止中	運転状態	ワーク排出ロボットが作業停止状態のときONとなる。
加工ロボット停止中	運転状態	加工ロボット（塗装ロボット）が作業停止中のときONとなる。
コンベヤ作動中	運転状態	コンベヤがワーク搬送中のときONとなる。
ワーク残	システムデータ	加工すべきワークの残数を記憶する。

【００６８】
他の接続パターンとして、図８のワーク排出ロボット用作動入力パターンがある。これは、ロボット作動のひとつで、ワークが到着した状態で、前工程ロボット（塗装ロボット）が作業停止中の場合に作動するロジックである。各パターン変数名の具体的な中身は、下記の表１１に示すようになっている。
【００６９】
【表１１】

パターン変数名	接続変数種別	変数説明
電源ON	システムデータ	システムの電源がON状態のときONとなる。
ワーク到着	ワーク検出	ワークがロボット作業位置に到着したときにONとなる。
ワーク停止中	運転状態	ワークがコンベヤ上で停止した状態のときONとなる。
前工程ロボット停止中	運転状態	前工程ロボット（塗装ロボット）が作業停止中のときONとなる。

【００７０】
さらに他の接続パターンとして、ワーク機種設定パターンがある。これは、ロボットに対してワーク機種データを設定するためのパターンであり、設定タイミングがＯＦＦからＯＮになったタイミングで設定する。各パターン変数名の具体的な中身は、下記の表１２に示すようになっている。
【００７１】
【表１２】

パターン変数名	接続変数種別	変数説明
設定タイミング	作動指令	データ設定タイミングのときONとなる。
機種データ	ワークデータ	ワーク機種データ。

【００７２】
他の接続パターンとして、作業停止条件パターンがある。これは図１３の中央位置に示さ

れたものと同じものである。これは、ロボットへの強制作業停止入力用パターンであり、異常発生中もしくは操作者からの緊急停止入力があった場合に作動する。各パターン変数名の具体的な中身は、下記の表１３に示すようになっている。

【００７３】

【表１３】

パターン変数名	接続変数種別	変数説明
異常発生中	異常データ	ライン異常発生中の場合にONとなる。
緊急停止入力	システムデータ	操作者からの緊急停止入力があった場合ONとなる。

【００７４】

上記した各処理を実施することにより、ラダー部品とそれを接続する接続パターンが決定されるとともに、各入出力Ｉ／Ｆのリンクを設定したならば、ラダー生成部１８がラダー部品とパターンからシステムのラダープログラムを構築する。つまり、設計情報とラダー部品，接続パターンを基にラダー言語で記述される制御プログラム（システムラダー）を構築するとともに物理アドレスの自動割付を行う。そして、具体的には図１６に示すフローチャートを実施するようになる。

【００７５】

すなわち、まず、ユーザが登録した利用部品のリストを利用部品情報登録部１４から取得する（ＳＴ２１）。次いで、各ラダー部品について入力Ｉ／Ｆのリストを利用部品情報登録部１４から取得する（ＳＴ２２）。さらに、取得したリストにある各入力Ｉ／Ｆについて割り付けられたパターン情報を、接続パターン情報取得部１５から取得する（ＳＴ２３）。

【００７６】

次に、パターン内変数名をリンク設定された他部品出力Ｉ／Ｆ変数名に置き換える（ＳＴ２４）。また、部品入力Ｉ／Ｆへの受け渡しデータとなる接続パターンの出力変数（Ｏｕｔ＿Ｐａｔｔｅｒｎ変数）を割付先の部品入力Ｉ／Ｆ変数名に置き換える。

【００７７】

その後、ラダー生成結果として、作成したパターンラダーをラダープログラムファイルに格納する（ＳＴ２５）。そして、上記した処理をラダー部品の全ての入力Ｉ／Ｆについて行う（ＳＴ２６）。

【００７８】

一方、全ての入力Ｉ／Ｆに対して処理が完了したならば（ステップ２６でＹｅｓ）、ラダー部品管理部１１からラダー部品を取得し、生成ラダーファイルへ保存する（ＳＴ２７）。すなわち、部品のラダーをラダー部品管理部１１より取得し、そのままラダープログラムファイルへ出力する。そして、上記した処理を登録された全てのラダー部品の全て行う（ＳＴ２８）。

【００７９】

次いで、部品ラダーの全変数について物理アドレス自動割付を行う。すなわち、割付結果を生成ラダーファイルへ保存する。そして、部品ラダーで使われる全変数について物理アドレスを自動で割り付け、割付結果をラダープログラムファイルへ出力する（ＳＴ２９）。

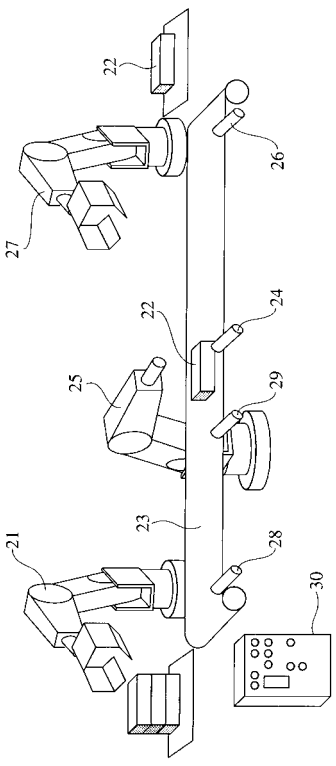
【００８０】

これにより、パターン入力／出力変数名を接続設定された他部品のＩ／Ｆ変数名へ名称変更する方法により合成時のリンクを解決する。そして、係るラダー生成を行い、最終的に生成された制御プログラムの一例を示すと、図１７のようになる。

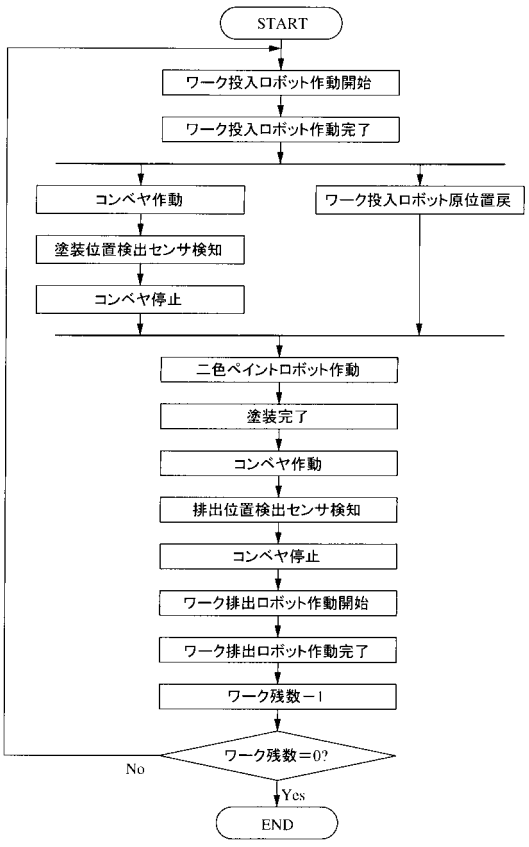
【００８１】
すなわち、部品ごとに接続パターン（ラダー）を配置する。そして、接続パターンで入力変数（接点変数）を、設定されたパターン接続先の他部品（ラダー部品）の出力Ｉ／Ｆ変数に置き換える（▲１▼）。次いで、パターン出力変数（コイル変数）をパターンを割り付けたラダー部品の入力Ｉ／Ｆ変数へ置き換える（▲２▼）。また、パターン内独自変数（パターン内でのみ使われる変数）はそのまま用いる（▲３▼）。そして、全ての入力Ｉ／Ｆに対するパターンを配置した後、ラダー部品自体のラダーを配置する（▲４▼）。なお、部品ラダーの変更は行わずそのまま配置する。以後、上記処理を繰り返すことにより、全ての部品のラダープログラムを作成することができる。
【００８２】
【発明の効果】
以上のように、この発明では、ラダー部品とそれを接続する接続パターンを用意し、ユーザは、利用するラダー部品を選ぶことにより、自動的或いは半自動的（リストアップされた候補の中から選択する）に接続パターンや、変数のリンク設定等がなされるので、生産設備、及びその制御プログラムを適切な単位で部品化して再利用することかでき、効率的な開発が行え、信頼性の高いプログラムを作成することができる。
【図面の簡単な説明】
【図１】本発明を説明するためのシステム例である。
【図２】図１のシステムのワークフローを示したフローチャートである。
【図３】図１のシステムをこれまでの方法で作成したラダープログラムの一例である。
【図４】本発明で使用するラダー部品「ピッキングロボット」のラダープログラムである。
【図５】本発明で使用するラダー部品「塗装ロボット」のラダープログラムである。
【図６】本発明で使用するラダー部品「コンベヤ」のラダープログラムである。
【図７】本発明で使用する接続パターン「ロボット作動（ワーク残チェック有）」のラダープログラムである。
【図８】本発明で使用する接続パターン「ロボット作動（ワーク到着／前ロボット停止）」のラダープログラムである。
【図９】本発明で使用する接続パターン「ワーク機種設定」のラダープログラムである。
【図１０】本発明で使用する接続パターン「作業停止条件１」のラダープログラムである。
【図１１】本発明の好適な一実施の形態を示す図である。
【図１２】入出力Ｉ／Ｆのリンクの設定処理の一例を示す図である。
【図１３】図１２を詳細に示した図である。
【図１４】ラダー部品並びに接続パターンを選択し、接続する処理機能を説明するフローチャートである。
【図１５】図４に示すステップ７の部品入力Ｉ／Ｆへのパターン割付処理の具体的なアルゴリズムを示すフローチャートである。
【図１６】プログラム生成部の機能を示すフローチャートである。
【図１７】本発明の方法で作成したラダープログラムの具体例である。
【図１８】接続パターン管理部のデータ構造の一例を示す図である。
【図１９】ラダー部品管理部のデータ構造の一例を示す図である。
【図２０】部品Ｉ／Ｆリストの一例を示す図である。
【図２１】接続パターン設定画面の基となるデータの構成を示す図である。
【符号の説明】
１０　ツール（制御プログラム作成装置）
１１　ラダー部品管理部
１２　接続パターン管理部
１３　接続情報入力部
１４　利用部品情報管理部

１５　接続パターン情報取得部
１６　Ｉ／Ｆ変数情報取得部
１７　パターン自動割付部
１８　ラダー生成部

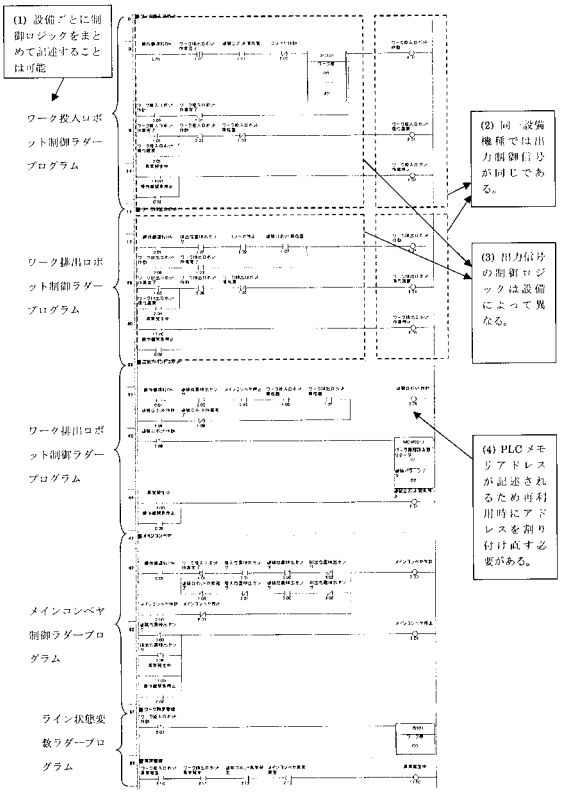
【図１】



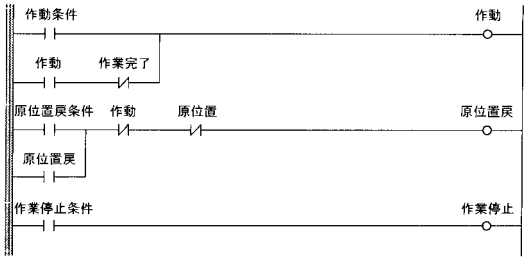
【図２】



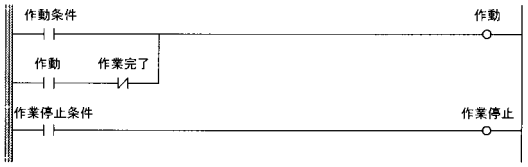
【図３】



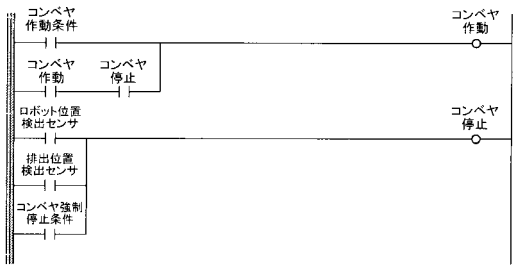
【図４】



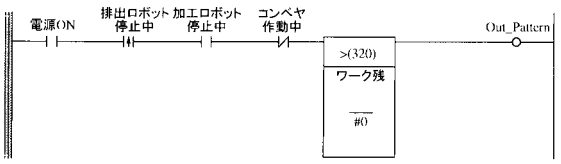
【図５】



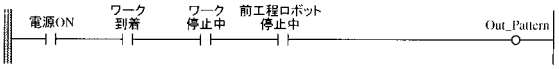
【図６】



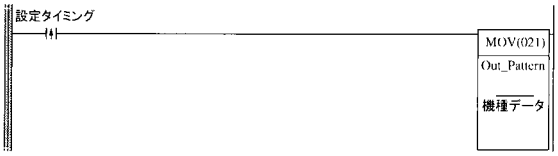
【図７】



【図８】



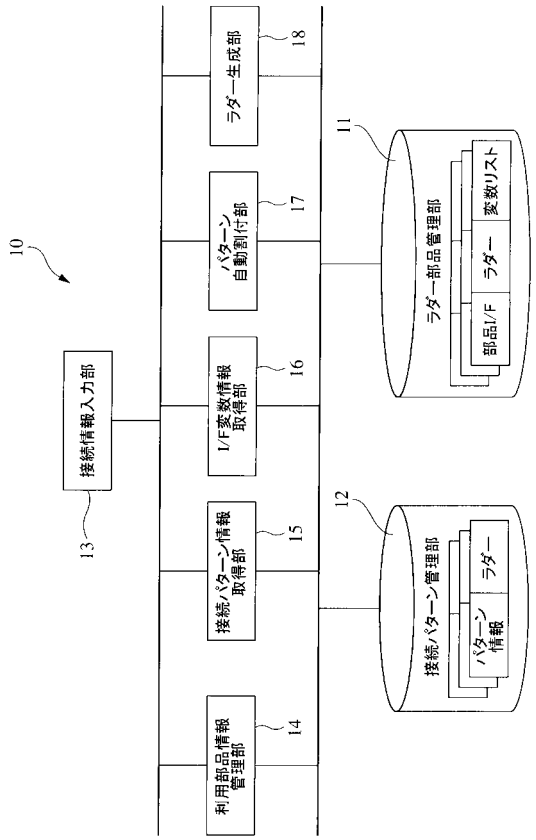
【図９】



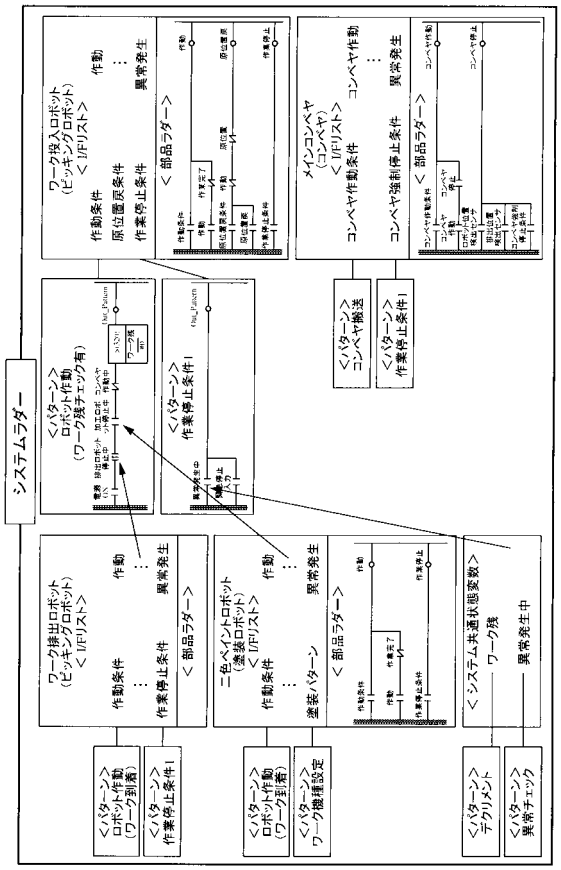
【図１０】



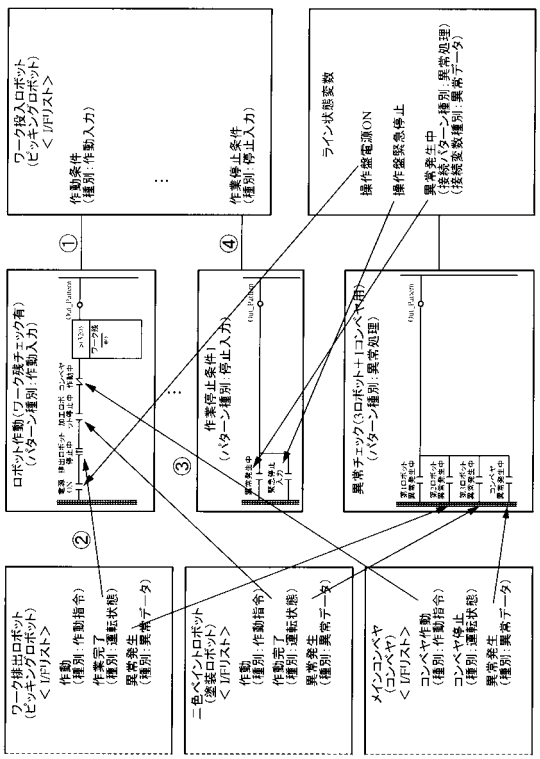
【図１１】



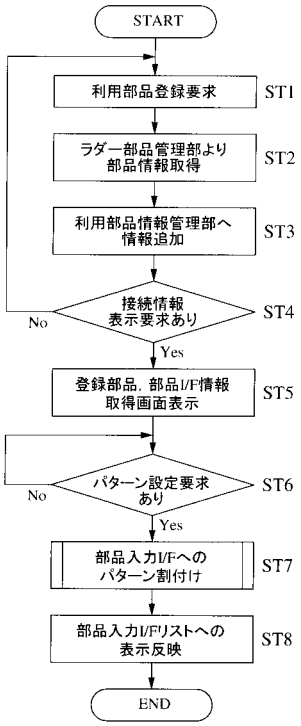
【図１２】



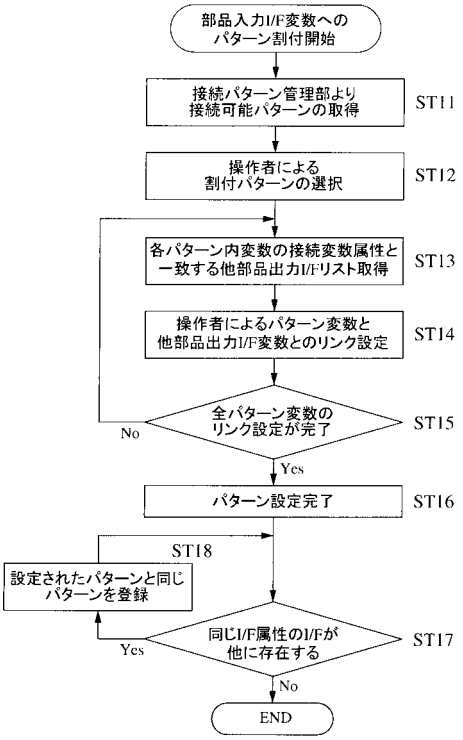
【図１３】



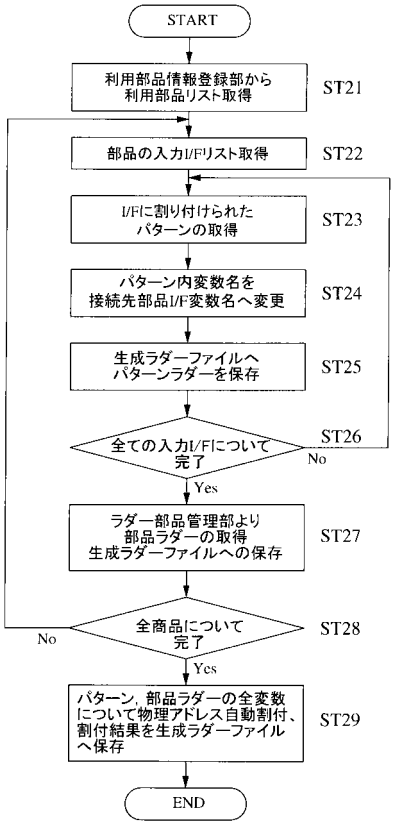
【図１４】



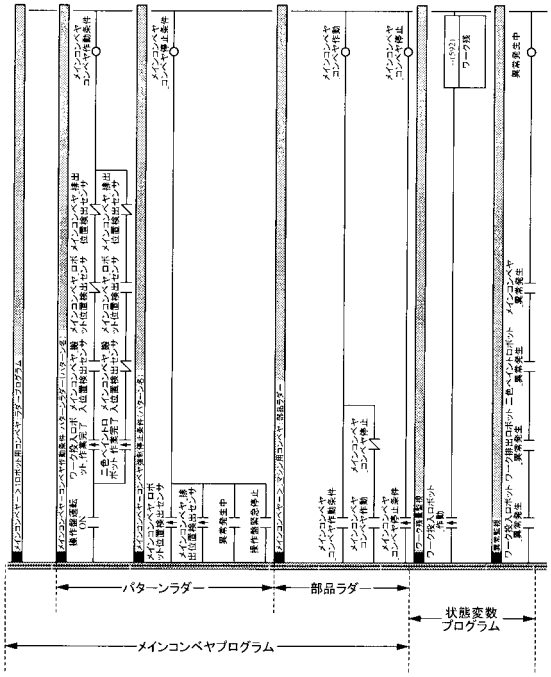
【図１５】



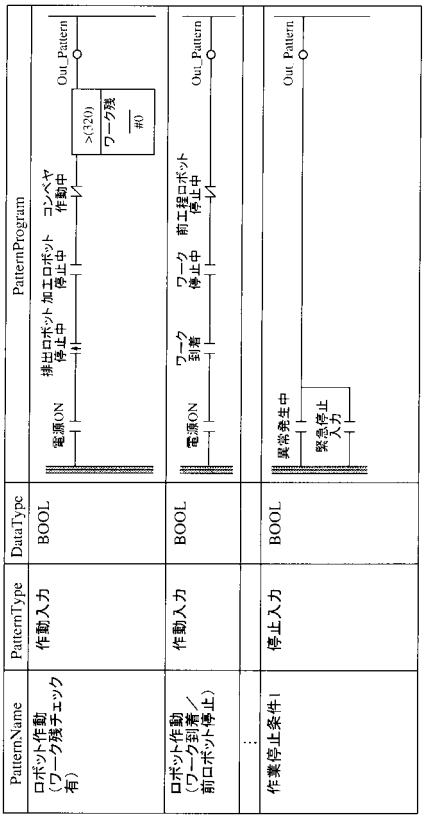
【図１６】



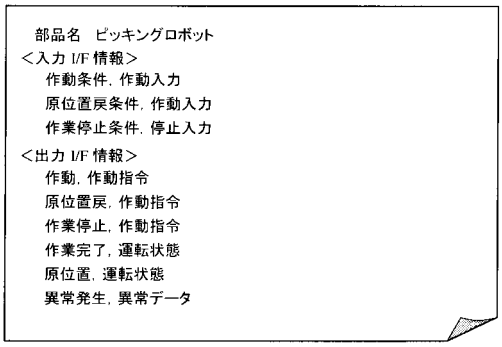
【図１７】



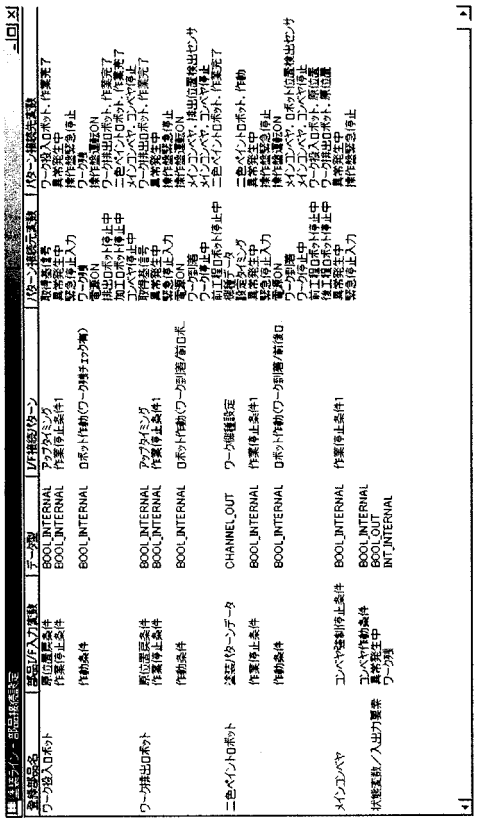
【図１８】



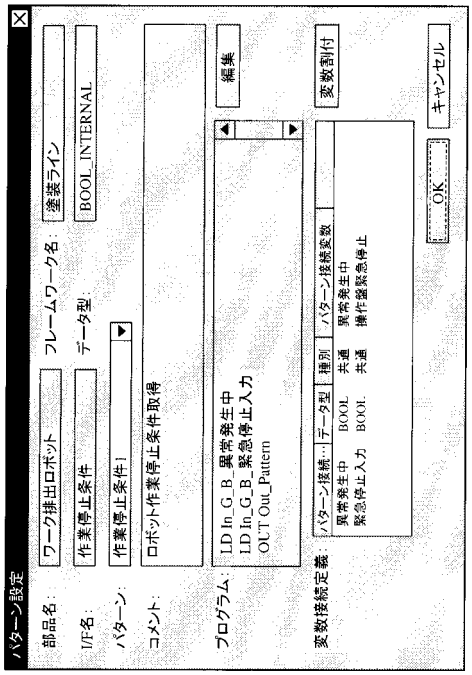
【図１９】


部品名　ピッキングロボット
＜入力 I/F 情報＞
作動条件，作動入力
原位置戻条件，作動入力
作業停止条件，停止入力
＜出力 I/F 情報＞
作動，作動指令
原位置戻，作動指令
作業停止，作動指令
作業完了，運転状態
原位置，運転状態
異常発生，異常データ

【図２０】



【図２１】


パターン設定
部品名：
ワーク排出ロボット
フレームワーク名：
塗装ライン
I/F名：
作業停止条件
データ型：
BOOL_INTERNAL
パターン：
作業停止条件1
コメント：
ロボット作業停止条件取得
プログラム：
LD In_G_B_異常発生中
LD In_G_B_緊急停止入力
OUT Out_Pattern
編集
変数接続定義：
パターン接続：
データ型
種別
パターン接続変数
異常発生中　BOOL　共通　異常発生中
緊急停止入力　BOOL　共通　操作盤緊急停止
変数割付
OK
キャンセル