

オブジェクト指向と状態遷移モデルに基づく 6M-6 シーケンス制御用言語のグラフィックエディタ

松田晃典、酒井充、米田政明、長谷博行
富山大学工学部電子情報工学科

1. はじめに

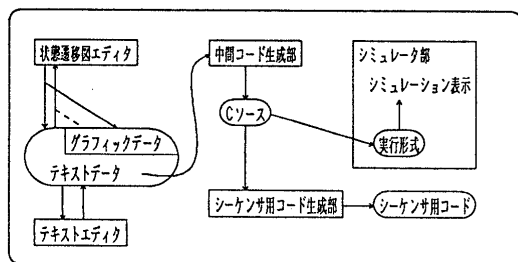
現在シーケンス制御用言語で主に利用されているのはラダーダイアグラム方式であるが、様々な問題点を持っている。そこで本研究では、オブジェクト指向と状態遷移モデルを採用した新しい言語を作成した^[1]。本言語によりプログラムの部品化、並列動作のわかりやすい記述が可能となった。さらにインタラクティブなプログラミングが行えるエディタや、容易に動作解析ができるように、タイムチャート出力や状態遷移が把握できるシミュレータなどの統合環境が重要となってくる。

今回はXウィンドウシステム上で動作する状態遷移図エディタを作成したので報告する。

2. シーケンス制御支援システムの構成

本システムの構成要素を以下に示す。

- ・状態遷移図エディタ：今回の報告の中心となるウィンドウ上に状態遷移図を描くことで本言語のソースを作成でき、また状態遷移図への変換ができる。
- ・テキストエディタ：本言語のソースを編集する。
- ・中間コード生成部：テキストデータである本言語のソースからトップクラスとなるものを選択し、クラスの階層構造から全てのテキストデータを連結し、本研究で開発したコンパイラに通すことで中間コードであるCのソースを生成する。
- ・シーケンサ用コード生成部：中間コードからシーケンサ用のコードに変換する。
- ・シミュレータ部：プログラム上では状態の変化がわかりにくいので、その様子をグラフィックで表わすことで、クラスやオブジェクトの階層構造、オブジェクトの状態遷移が的確に表示される。



全体のシステム構成

A Graphical Editor for Sequence Control Language
Based on Object-Oriented and State-Transition Models
Akinori Matsuda, Mitsuru Sakai, Masaaki Yoneda, Hiroyuki Iase
Toyama Univ.

3. 状態遷移図エディタ作成における基本方針

エディタは以下のことに注意して作成した。

- ・本言語の予約語の意味さえ知っていれば操作できるようにする。
- ・わかりやすく、かつ少ない操作で入力できるようにする。
- ・メニューやボタン類はわかりやすく、できるだけ少なくする。
- ・ウィンドウの数が多いと重なり合いわかりづらくなるので必要最小限にする。
- ・どの大きさのディスプレイにも対応できるように、ウィンドウマネージャーを用いた弾力的なものとする。
- ・エディタでプログラミングしていく場合、オブジェクトベースで行った方が自然かもしれないが、作成するプログラムの、コード量を抑え、ライブラリとして保存し再利用すること等を考えるとクラスを記述していく方が利点が大いのでクラスベースで行う。

次に状態遷移図エディタの説明のために必要となるオブジェクト指向、状態遷移モデルについて簡単に説明する。

4. オブジェクト指向

オブジェクト指向では、オブジェクトやクラス、インスタンスなどの用語が使われるが、ここではオブジェクトとは部品のような具体的なポート番号を持つスイッチなどを意味する。クラスとはオブジェクトが持つ共通の性質をまとめたものであり、プログラムの設計図である。インスタンスはクラスの中で定義される仮のオブジェクトである。

制御対象となる機械やシステムをオブジェクトとしてとらえる。またそれらを構成する動作部品などもオブジェクトとしてとらえ、その部品はさらに内部に部品を持っているので制御対象はオブジェクトの階層構造を持つとみなせる。又、各オブジェクトはメッセージ通信^[1]を行う。

5. 状態遷移モデル

制御対象としてのオブジェクトは多くの場合内部状態を持つと考えることができ、メッセージを受けると対応する動作を行い、必要に応じて内部状態を変える。つまり、オブジェクト内では逐次動作である。しかし、実際は機械資源を有効に利用するためパイプライン処理が存在する。そこで、状態の階層化を用い副状態遷移の1つとしてパイプライン処理を簡単に記述できるようにした。

6. 状態遷移図エディタ

システムを起動するとまず基本メニューウィンドウが表示される。このウィンドウメニューの一つであるグラフィック編集を選択すると主状態ウィンドウが開き、状態遷移図

エディタが開始する。ここからさらに副状態ウィンドウ、パイプラインウィンドウ、ノードウィンドウを選択できる。

また既存ファイルからロードするとき、テキストデータを選択するが、付随するグラフィックデータがあれば自動的にロードされる。

6. 1 主状態ウィンドウ

このウィンドウのメニューはセーブ、終了、ヘルプからなる。セーブすると本言語のソースであるテキストデータとグラフィックデータが作成される。

メニューの下にはクラス名の入力、そのクラスで使う変数宣言部がある。この変数宣言は状態遷移を入力していくときに必要となるものでこれがないと、メッセージの送信や受信ができないことになる。

次に関数型メッセージの動作プログラムの入力用ウィンドウと、その下に状態遷移の入力用ウィンドウがある。

状態遷移の入力には状態を表す○を配置する。それに伴い後述のノードウィンドウが開くので、そこでノード名やメッセージ送信を入力する。

6. 2 副状態ウィンドウ

副状態遷移を記述するウィンドウである。このウィンドウは副状態遷移名と、状態遷移図の2つの入力用ウィンドウから構成される。

6. 3 パイプラインウィンドウ

副状態遷移の一つであるパイプライン処理を記述するウィンドウである。このウィンドウは副状態ウィンドウに、排他条件の入力を付け加えたものからなる。

6. 4 ノードウィンドウ

これはノードの動作を記述するウィンドウである。ウィンドウにはノード名、ノードの中だけで有効なローカル宣言、動作の記述のそれぞれの入力ウィンドウからなる。

動作の記述には入力、変更、消去メニューを使い、その結果がstatement_listの下にウィンドウに表示される。

入力、変更メニューともサブメニューとして送信、受信、遷移、EXIT、代入式、その他のメニューがある。

送信メニューは内外オブジェクトへのメッセージ送信に使い、受信メニューはメッセージを判断する条件文を入力するために使う。遷移メニューには更にノード、副状態、パイプラインのサブメニューがある。ノードメニューはノードからノードへの矢印を状態遷移ウィンドウに描くために使う。また副状態、パイプラインのそれぞれのメニューは副状態ウィンドウ、パイプラインウィンドウを新たに開くために使う。EXITメニューは副状態（パイプラインを含む）から主状態への戻りを入力するためのものである。

最後に自動サイロシステム^[2]におけるトップクラスを状態遷移図エディタを用いて記述した例を示す。

7. まとめ

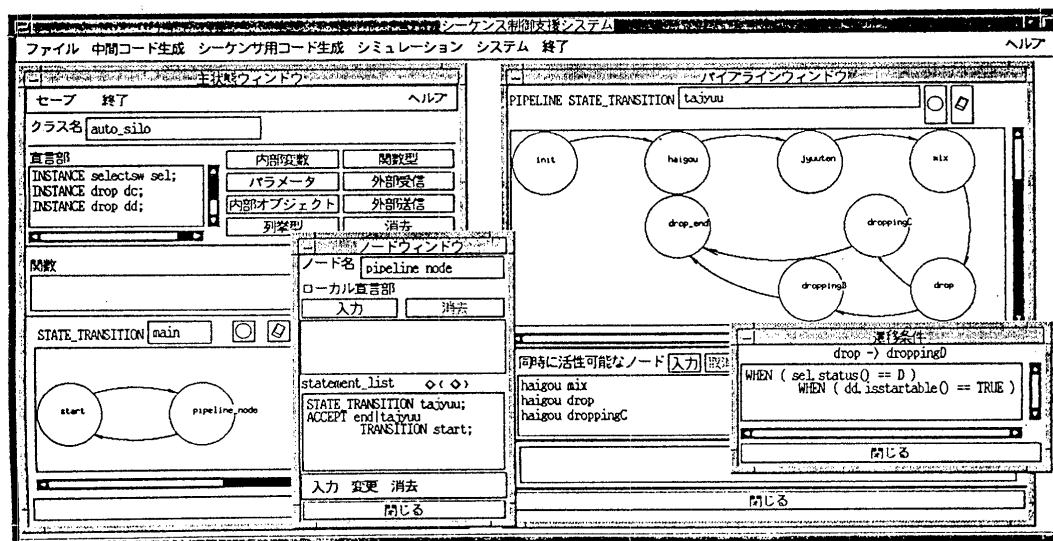
今回はウィンドウ上でインタラクティブに状態遷移をプログラムできる状態遷移図エディタの報告を行った。これにより遷移が図に示され、わかりやすく制御対象を記述できる。また予約語であるキーワードをほとんどマウスで選択し入力するため容易にプログラミングできると予想される。

今後システム構成に示したように、オブジェクトの動作状態のグラフィック化でシーケンス制御で多く見られる並列動作の確認が容易となるシミュレータやタイムチャート表示、さらにわかりやすさのためアニメーション表示などの支援ツールを開発していかなければならない。

参考文献

[1] 滝田、酒井、米田、長谷、松田：オブジェクト指向と状態遷移モデルによるシーケンス制御用言語、情報処理学会研究報告、92-SE-83、PP.139-146

[2] 電気学会：シーケンス制御工学、オーム社、(1988)



自動サイロシステムにおける適用例