

シーケンス制御用プログラムの 統合化開発環境の試作

古田学、米田政明、長谷博行、酒井充
(富山大学工学部電子情報工学科)

7T-2

1 はじめに

シーケンス制御は機械や工場などの自動制御に広く用いられている制御方法である。現在のシーケンス制御技術はプログラマブルコントローラを中心にしたものであり、多くの制御機能をソフトウェアで実現することが可能である。ソフトウェアを記述するための言語として、主にラダーダイアグラム方式が利用されているが、様々な問題点を持っている。そこで本研究室ではオブジェクト指向と状態遷移モデルを取り入れた新しい言語（以下、本言語）を開発した⁽¹⁾。これまでに、本言語開発のためのコンパイラや本言語によるプログラミングを支援するための様々なツールが作成された。今回はそれらのツールを1つのシステム上で動作させるための統合化開発環境を試作したので報告する。

2 本言語の特徴

本言語ではオブジェクト指向におけるクラスとオブジェクトの概念を取り入れている。制御対象となる機械やシステム、さらにはそれらを構成する動作部品をオブジェクトとして捉え、オブジェクトが持つ共通の性質をまとめたものをクラスとしている。また、状態遷移モデルによって個々のオブジェクトの動作記述の枠組みを与えている。

ラダー図と比較して次のような特徴がある。

- ・モジュール化、部品化が可能。
- ・シーケンスの状態遷移を分かりやすく表現することが可能。
- ・可読性に富み、保守や修正が容易。

3 各開発支援ツールの概要

本システムが統合している各開発支援ツールについての説明を行う。

・C言語へのコンパイラ

本言語ソースプログラムをCプログラムに変換する。ここで作成されたCプログラムをCコンパイラでコンパイルして得られる実行形式プログラム（以

下、実行形式プログラム）により、本言語の動作を確認することができる。

・IL言語へのコンパイラ

本言語ソースプログラムをILプログラムに変換する。IL言語はプログラマブルコントローラの内部表現と一対一に対応するシーケンス制御用言語である。

・グラフィックエディタ

ウィンドウ上に状態遷移図を描くことで本言語ソースプログラムを作成でき、またテキストエディタで作成した本言語ソースプログラムから状態遷移図への変換ができる。状態遷移をグラフィックで表示、編集することができるため、状態遷移の様子が分かりやすくなり、プログラミングやデバッグを行いやすくなるという利点がある（図1）。

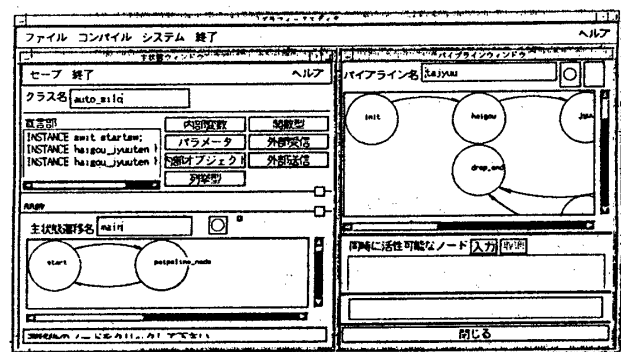


図1 グラフィックエディタ

・ILEミュレータ

ILプログラムのエミュレーションを行う。これにより、実際に制御装置を動作させる前にILプログラムの動作確認を行うことができる。

・タイムチャート表示ツール

実行形式プログラムの出力、もしくは、ILEミュレータの出力をタイムチャート表示する。これによりプログラムの動作解析を視覚的に行うことが可能となる（図2）。

・状態遷移図表示ツール

実行形式プログラムの出力を状態遷移図表示する。タイムチャート表示ツールと同様、プログラムの動作解析を視覚的に行うことができる（図3）。

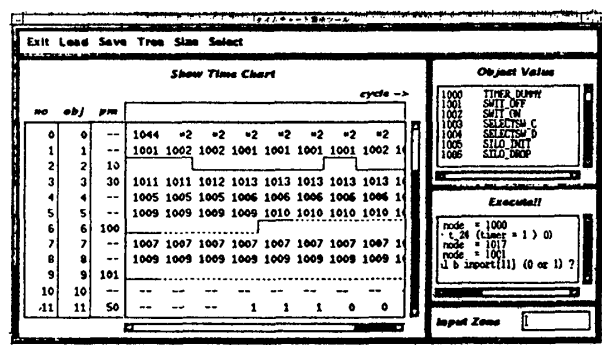


図2 タイムチャート表示ツール

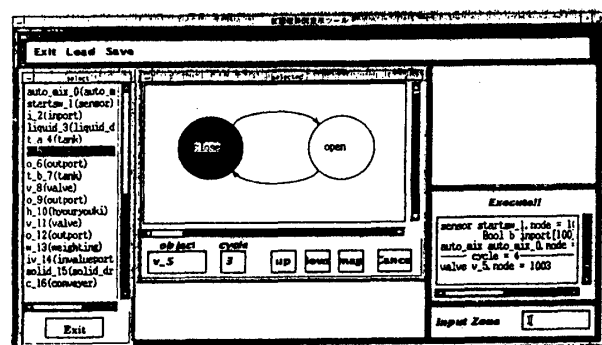


図3 状態遷移図表示ツール

4 本システムの概要

4-1 設計方針

これまで、上記の各ツールを個別に実行していたため、様々な面倒な作業が必要であった。そのような状況を解決するために、各ツールを一貫して操作することのできる統合環境を提供する本システムが作成された。つまり、ユーザが本言語によるプログラミングを行いやすいシステムを作る、ということが目的である。

また、ユーザが操作しやすいシステムにすることも重要である。本システムはXウィンドウシステム上で動作する。ユーザインタフェースをプログラムするとき、Xlibだけではプログラミングが困難となりコード量が大きくなるためOSF/Motifを使用している。これにより、ウィンドウの操作が統一され、ユーザにとって操作しやすいシステムとなる。

4-2 実行手順

本システムを起動すると図4のメニューウィンドウが開く。

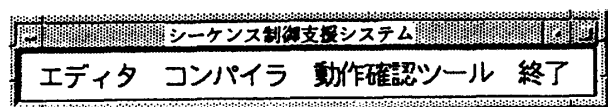


図4 メニューウィンドウ

各ツールはメニューから選択することにより実行される。メニューの階層関係を以下に示す。

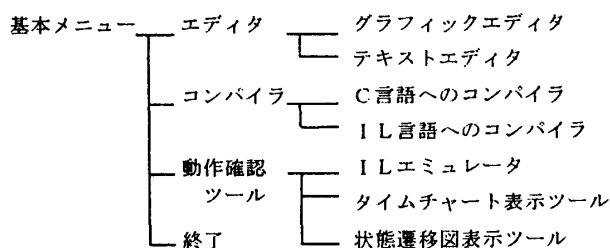


図5 メニューの階層関係

本システムにおける各ツールの処理の流れを以下に示す。図中、二重枠の矩形で囲まれた部分は本研究で開発したツールを表し、矩形で囲まれた部分は市販のツール、それ以外はデータを表している。

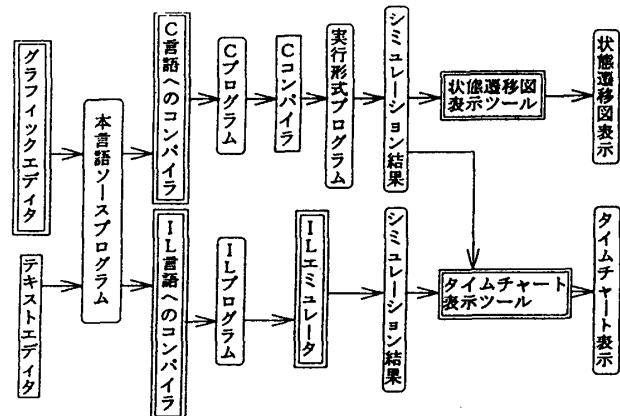


図6 システム構成

5 まとめ

各開発支援ツールを統合化したことにより、本言語によるプログラミングからデバッグまでの一連の作業をスムーズに行うことができるようになった。

ウィンドウシステムにOSF/Motifを使用しているため、すべての操作が一定の規則に基づいており一貫性がある。これにより、ユーザがシステムに近付きやすいものとなった。

今後は細部の表示や操作方法を改善することによって、さらにユーザフレンドリなものにしたいと考えている。

参考文献

- (1) 酒井、酒井、米田、長谷：「オブジェクト指向に基づくシーケンス制御用言語とI L言語へのコンパイラの作成」、情報研報、COMP94-99, SS94-56(1995-03)