

# SDDサポートシステム SDD Support System

福島 弘道                      杉本 正勝                      鞠嗣 直哉  
FUKUSHIMA                      SUGIMOTO                      UKAI  
Hiromichi                      Masakatsu                      Naoya

(富士通株式会社)

Fujitsu Limited

## 1. はじめに

ソフトウェアは「人間の考えを記述したもの」である。それ故、ソフトウェアの設計者・作成者の発想なり考え方が自然の形で盛り込めるような構造化されたドキュメントが必要となる。我々はこの要求に対しSDD図面を用い、人間の視覚の役割を十分に活用する方式を提案した。<sup>(1),(2)</sup>

SDDとは、ソフトウェアの図形表示技術を意味し、その目的は次の通りである。

- (1) 高品質のソフトウェアの設計・作成。
- (2) ソフトウェアの生産性・保守性の向上。

本稿で報告するSDDサポートシステムで対象にする図面は、図-1の図面(c)の部分に関するものである。

SDDサポートシステムとは、SDD図面を使用した『ソフトウェア開発用のサポートシステム』であり、SDD図面の入力、編集、チェック、出力、保守及び実行を行う。

## 2. SDD図面

SDD図面は、図-2に示すボックスをプログラムの実

行の順に直線で結合したものであり、通常のフローチャートとソースプログラムが結合された形で、プログラムの構造、論理の流れが直感的に理解できる。更に、SDD図面だけがコンピュータに対する入力情報となるため、従来問題になっていた「フローチャートとソースプログラムの不一致」と言った不都合は解消される。

1枚のSDD図面に表現出来るボックスの数は最高120個で、各ボックスには注釈、位置、高級プログラミング言語の式、文及び接続等のボックス情報を記述する。

## 3. SDDサポートシステムの構成

SDDサポートシステムは図面入力部、図面出力部、図面エディタ部、プログラム処理(トランスフォーマ)部からなり、FACOM M-180IIをホストとし、TSSで運用される。

### 3.1 図面の入力

「SDD LOGIC DIAGRAM」用紙を使用して、手書きSDD図面を作成する。その図面は、下記のいずれかの方法でコンピュータに入力される。

#### (1) 対話型入力

対話型図面入力プログラムを起動すると、ディスプレイ端末に図面のデザイン・データに関する問い合わせがあり、必要情報を答える。次にボックスに関する問い合わせがあり、使用ボックスのタイプを指定すると、そのボックス・イメージの図がディスプレイに現れる。

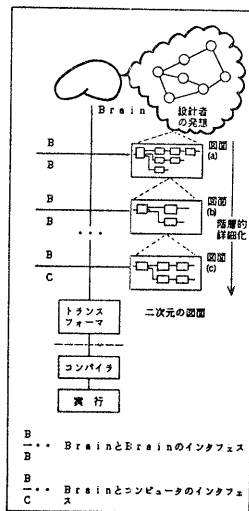
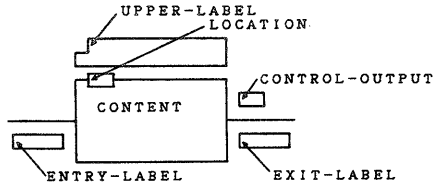


図-1 SDDサポートシステムによるソフトウェアの設計・開発



Control Structure	Representation in Higher Level Programming Language	Representation in SDD
Sequential Execution	U; V;	(Single-box)  Several statements can be placed in a box.
IF	IF C THEN U1; ELSE U2; V;	(IF-box) 
Case	SELECT (CID); WHEN (CID=1) U1; WHEN (CID=2) U2; WHEN (CID=n) Un; END;	(Case-box) 
Repetition	DO I=1 TO N BY 2; U; V; END; W;	(DO-box) 
	DO UNTIL (P); U; V; END; W;	(DO_UNTIL-box) 

図一 2 SDD 図面の基本ボックスとボックスのタイプ

その図に従って必要情報を入力すれば、1ボックスの登録が終了する。必要なボックス及びボックス情報全ての登録が終れば、1図面のファイルが完成する。

## (2) バッチ入力

手書きSDD図面をもとにして、カードデッキを作成する。そのカードデッキをバッチ図面入力処理プログラムにかければ、(1) 対話型入力で作成されるのと同じ形式のファイルが完成する。

## 3. 2 図面の出力

図面出力プログラムは、登録された手書きSDD図面を

プリンタに出力する。この図面に対してデバッグを行い、デバッグ後は保守図面となる。

## 3. 3 トランスフォーマ

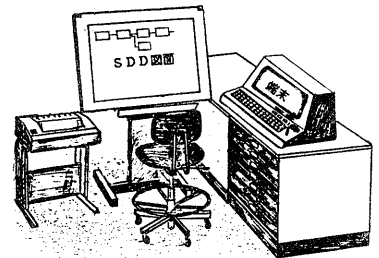
トランスフォーマは、登録された図面と、必要に応じて他のモジュールとをマージし、それをコンパイラのソース・コードに変換する。

## 3. 4 デバッグ・ツール

トランスフォーマ開始時に、実行されるボックスの制御の流れを表示する、といったデバッグ・オプションを指示できる。

## 4. SDDサポートシステムの利用環境

SDDサポートシステムは、SDD図面設計ボード、ディスプレイ端末、プリンタより構成され、図一3に示すような環境で利用される。



### (1) 論理を追う

時は、SDD図面設計ボード上のSDD図面を見る。

### (2) デバッグは、ディスプレイ

端末で対話型で行う。

### (3) プログラム論理の変更は、ディスプレイ端末を使用。

### (4) 変更後の図面は、すぐその場で印刷出来れば良いが、例えば、デバッグの切れ目とか、1日単位で変更後の図面を出すという形態でも十分実用になると考える。

## 5. SDDサポートシステムの適用例

SDDサポートシステムは実験システムとして、『OASYS 100』（日本語タイプライタ）のソフトウェアの開発に適用された。またSDDサポートシステムの開発にも適用している。SDDの適用に関し、担当者の意見を



まとめると次のようになる。

- (1) 図面でプログラムの論理がよく見渡せるので、冗長のない、すっきりとした論理が組める。
- (2) SDD図面を使えば、過去に作成したプログラム（例えば、3か月前に作成したプログラム）の論理を思い出すのが容易になる。
- (3) プログラムのリストを端末に表示して論理を追うのは、目が疲れて作業がはかどらないことがある。しかし、SDDではそれが無い。

これらのことからプログラムの作成、テスト、メンテナンスの分業がより容易になり、各段階の作業を楽にすることが出来る。

## 6. 考察

SDDサポートシステムの開発を進めて行く間で、多くの人からSDDによせられた、意見・提案・要望をまとめると次の通りになる。

- (1) ソフトウェアの部品化、共通化については、いまだ実現していないが、今後うまく行くのか？
  - SDDの論理の見易い図面で、大きく前進する。
- (2) ソフトウェアの部品化にともない、部品がブラックボックス化して行くのではないかと？
  - ハードウェアの論理を表現するのに回路図が必要であるように、ソフトウェアにおいても、それに対応するような図面が今後とも必要になる。部品化が進んで行く時、その部品を説明する機能仕様は必要であり、その表現にも図面形式が良いと考える。
- (3) 通常のプログラミングとSDDとの比較は？
  - 現時点におけるSDD表現は、通常のプログラミング言語のレベルと同じであるが、今後、仕様レベルにまで拡張して行きたい。
- (4) プリプロセッサやトランスフォーマ等を通して、既存のコンパイラを使うのは、階層が増えた分だけリスト量が増えデバッグが大変になるのではないかと？

- SDDを使えばバグの数が軽減され、リスト量の問題もなくなると考える。
- (5) マイコン用も含めて、素人でも簡単にソフトウェアを開発出来るようなツールに拡張して欲しい。
- (6) コメント部分は日本語表現の方が良い。

## 7. おわりに

ソフトウェアを二次元の構造化された図面として表すことにより、人間の視覚を十分に活用でき、図面を介して設計者・作成者・保守者の間で正確で、速やかな情報交換が可能となる。更にその上、ソフトウェアの部品図面としての定義も容易となる。今後、このSDDサポートシステムを大い充実させて行く積りである。

## 謝辞

SDDの提案者であるオフィス事業本部ソフトウェア事業部開発技術部神田部長及びいつも御指導頂いている開発部佐藤部長に深謝致します。

## 参考文献

- (1) SDD: ソフトウェアの図形表示技術  
神田 杉本 情報処理学会ソフトウェア工学研究会  
資料 13-3, 1980. 3. 17
- (2) Y. Kanda and M. Sugimoto,  
Software Diagram Description: SDD and Its Application, COMPSAC 80 Oct. 27-31, 1980
- (3) L. A. Belady, C. J. Evangelisti, and L. R. Power, GREEN-PRINT: A Graphic Representation of Structured Programs, IBM SYST J, VOL 19, NO 4, 1980