

工業用ディスプレイへのGKSの適用

3Q-6

木下 政利, 小山 深

(富士ファコム制御)

1. はじめに

グラフィックスに関する国際的な標準であるGKS (Graphical Kernel System) に関するインプリメント事例, マルチウィンドウへの適用事例などは, 今までも報告されている。

本稿では, 工業用ディスプレイの基本ソフトウェアとしてGKSを適用した例について紹介する。工業用グラフィックスは, それ特有の性質を持っているが, 特に要求されることは, 高速なレスポンス性である。GKSでの表示速度向上については, 従来からも議論がされて来ているが, 今回, 工業用ディスプレイ (IVC) の特徴 (画面の保存, フォーマット変換, グラフ表示機能など) を活かしたGKSとしてパッケージ化したので報告する。

2. 機能

工業用グラフィックパッケージ (FGKS) は, 一般的なグラフィックス装置に対してはGKSの仕様に準拠しており, “互換性”, “可搬性” を保証している。一方, 工業用ディスプレイに対して, 次のような機能を付加した。

- (1) 画面のディスプレイ内セグメントバッファへの初期ローディング
- (2) 画面の高速表示
- (3) GDP (Generalized drawing Primitive) ファンクション

FGKSの構成を図1に示す。

3. 実現方法

- (1) 画面のセグメントバッファへの初期ローディング
大量の画面 (固定画面) を保存する手段として, WISS (Workstation Independent Segment Storage) を用いている。CREATE-SEGMENTで指定された番号で画面をディスク内にシーケンシャルファイルと

して管理している。WISSファイルのデータは, 複数のワークステーション (表示装置) にコピーするなどの再利用が可能である。

工業用グラフィックスでは, 通常画面数が百枚以上にもなるので, WISSファイルは, 外部ファイルとして扱っているが, ディスクアクセスのために, 表示速度が遅くなる。この問題を解決するために, 頻繁に表示される画面, 高速に表示したい画面 (固定部) は, あらかじめディスプレイ装置側に登録して置き高速表示を図っている。

(2) ワークステーション記述表の作成

FGKSは, 各ワークステーションの能力を記述するワークステーション (WS) 記述表を, wscapという外部ファイルで定義できる。

これによりアプリケーションプログラムの変更をせずに属性などの変更が容易にできる。このwscap ファイルをあらかじめコマンドまたはシステム立ち上げ時に主記憶上にWS記述表として展開して置くことによりGKSの内部処理を少なくし高速化を図っている。

(3) 正規化変換の省略

正規化変換の処理は, 表示速度を遅くする原因となっている。FGKSではIVCに対して, この処理を省いてワールド座標系とデバイス座標系の1対1の対応をとっている。尚, WISSファイルへ保存する時は, 画面の互換性を重視し, 正規化変換された形式で記憶している。

(4) GDPファンクション

IVCの持つ機能を活かし次のようなGDPファンクションをサポートしている。

① 漢字出力

漢字の表示を行う。漢字の出力属性は, 文字列と同

Application of GKS to the Industrial display

Masatoshi Kinoshita, Fukashi Koyama

FUJIFACOM CORP.

じである。

② ユーザ・シンボルの登録

ユーザ・シンボル（マーカ）の登録は、あらかじめディスク内のファイルに8×8、16×16のドットパターンで定義して置きコマンドまたはシステム立ち上げ時にディスプレイ装置側に登録する。登録されたシンボルはPOLYMARKERファンクションで表示することができる。

ユーザ・シンボルの属性は、POLYMARKERと同じである。

③ 仮想画面

デバイス座標系の4倍の画面に表示を行う。画面移動はローカルに行う。

④ 数値表示

10進、16進、浮動小数点データなどの表示を行う。出力属性もデータとともに指定する。

⑤ グラフ表示

罫線、棒グラフ、折れ線グラフ、円グラフなどの表示を行う。

4. 使用方法

FGKSの使用方法を図2に示す。

- (1) wscap ファイルよりWS記述表を主記憶に展開する。このテーブルは、削除しない限り存在する。
- (2) Cまたはfortran 言語で作成されたプログラムを実行し、固定画面をセグメントデータとして保存する。
- (3) ユーザ・シンボルをエディター（vi）を用いてユーザ・シンボルファイルに作成する。
- (4) 固定画面、ユーザ・シンボルをディスプレイ装置に登録する。
- (5) アプリケーションでは、GKSの各種関数を用いてプログラムを作成する。

5. 性能

(1) 表示速度

正規化変換の省略、固定画面の初期ローディング、WS記述表の初期設定などを行うことにより高速表示が可能となった。UNIXの場合、ファイルが木構造で管理されているため、プログラムのローディング時間が計算機により異なるが、FGKSにおいては、コマンド入力から最初の表示までに1秒以内という高速表示を実現す

ることができた。

(2) メモリ

WS記述表は、wscap ファイルに定義されたWSタイプごとの情報を基にあらかじめ展開される。各プロセスは、このテーブルを共通に参照している。また、WS状態表は、必要なときにダイナミックに獲得しているためメモリ効率がよくなっている。

アプリケーションプログラムのオブジェクトサイズはGKSのライブラリを含めて約100Kバイトである。

6. おわりに

本稿では、工業用グラフィックスへのGKSの適用について紹介した。GKSが規定する仕様への整合性と、性能が重視されるインプリメントは相反する関係にあるが、今までに報告されている方式なども参考にして、期待する性能を得ることができた。

従来、工業用グラフィックスにおいては、性能重視のために専用言語による画面作成、制御が中心であったが今後要求されてくる標準化に対してGKSは、十分適用できることが判明し、これによってアプリケーション開発コストの低減、生産性の向上、再利用の向上が期待できる。

今後は、対話作画機能などを追加することにより、より充実したグラフィックパッケージにする考えである。

7. 参考文献

- (1) “Graphical Kernel System(GKS) Functional Description V7.2”, ISO/TC/97/SC5/WG2, Dec, 1982
- (2) “CYKERNEL解説書”, シニック, 1984

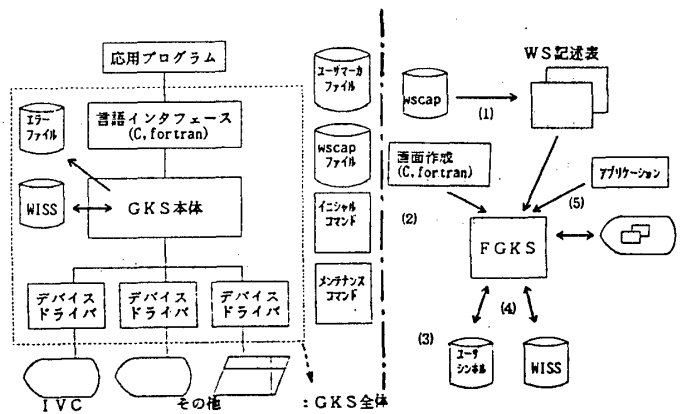


図1. FGKSの構成

図2. 使用方法概略図