新情報制御システム「MICREX-NX」のエンジニアリング支援ツール

竹内　良友（たけうち　よしとも）　山野　景章（やまの　けいしょう）　森　貴繁（もり　たかしげ）

1　まえがき

近年，日本国内におけるプロセスオートメーションシステムの市場は，新規需要が主体の成長期から，更新需要が主体の成熟期に遷移しており，市場の大きな伸びが期待できない状況が続いている。一方，プロセスオートメーションシステムは，そのハードウェア部分において，監視端末，コントローラ，入出力，それらをつなぐネットワークそれぞれについてのオープン化が進み，他社との差別化がしにくい状況になっている。このような市場環境下において，システムベンダーは，そのアプリケーションソフトウェアの高付加価値化，つまりシステムベンダーが技術の蓄積として保有する制御ノウハウの有効活用による他社との差別化に加え，成熟市場における激しい価格競争に勝ち抜くことのできる高効率・高品質なエンジニアリングを同時に実現せねばならないという難題を抱えている。

言い換えると，システムベンダーにとってエンジニアリングは重要性を増す一方であり，それを支援するツールは上記を解決するための適切なプラットフォームを提供しなければならない。

MICREX-NX のエンジニアリング支援ツールである ES（Engineering Station）は，先に述べたエンジニアリング支援ツールに対するニーズを満足すべく，さまざまな機能を提供する。

2　ES の機能と特徴

2.1　ES の機能

図1に ES の機能概観を示す。ES は，Windows〈注1〉パソコン上に構築され，HMI（Human Machine Interface），コントローラなど各種支援ツールの共通プラットフォームである。ES は複数のエンジニアリング用ソフトウェアの集合であり，それらすべてのソフトウェアは Manager によって統合管理されている。Manager は各種エンジニアリング支援ツールを呼び出し，ソフトウェア間のデータ連携や共通したプラットフォームにより支援ソフトウェアの管理を行う機能を提供する。

〈注1〉Windows：米国 Microsoft Corp. の登録商標

2.2　ES の特徴

(1)　統合されたエンジニアリングデータ

Manager から呼び出される各種エンジニアリング支援ツールのデータは，データベースによって統合管理されている。入力されたデータは，データベースにて一元化されるため，データを1か所で入力するだけで関連する複数のデータは自動更新される。これにより入力作業が効率化されるうえ，入力ミスなどのヒューマンエラーによるデータ不整合をなくし，ソフトウェア品質を向上させることができる。

(2)　階層化エンジニアリング

ES における，制御監視ソフトウェアなどのプロジェクトデータおよびライブラリデータは，Windows のエクスプローラライクな画面により階層化管理されている。これによりエンジニアは，直感的かつスピーディーに目的とするデータにアクセスすることができるとともに，階層単位のコピーアンドモディファイを容易に行うことができる。

(3)　View の構成

MICREX-NX エンジニアリングの中心となる三つの View は，それぞれの View が統合されたエンジニアリングデータを異なった角度から可視化する。エンジニアは自分が置かれた立場やそのフェーズによって，適宜 View を使い分けることができる。**図2**にその画面例を示す。

①　Component View

制御システムの AS（Automation System），AS に提供される I/O，OS（Operator Station），ES およびそれらをつなぐネットワークに関するハードウェア設定を行う。特徴としては，AS のリソース使用状況の監視を行うことができることが挙げられる。また，本 View から起動される HW Config（Hardware Configuration）は，既存の I/O をマイグレーションとして流用する場合にも対応しており，PDM（Process Device Manager）は，

竹内　良友

新情報制御システムの開発業務に従事。現在，富士電機システムズ(株)機器本部制御統括部商品開発部マネージャー。

山野　景章

新情報制御システムの開発業務に従事。現在，富士電機システムズ(株)機器本部制御統括部商品開発部。

森　貴繁

新情報制御システムの開発・企画業務に従事。現在，富士電機システムズ(株)機器本部制御統括部商品開発部。

図１　ES の機能概観

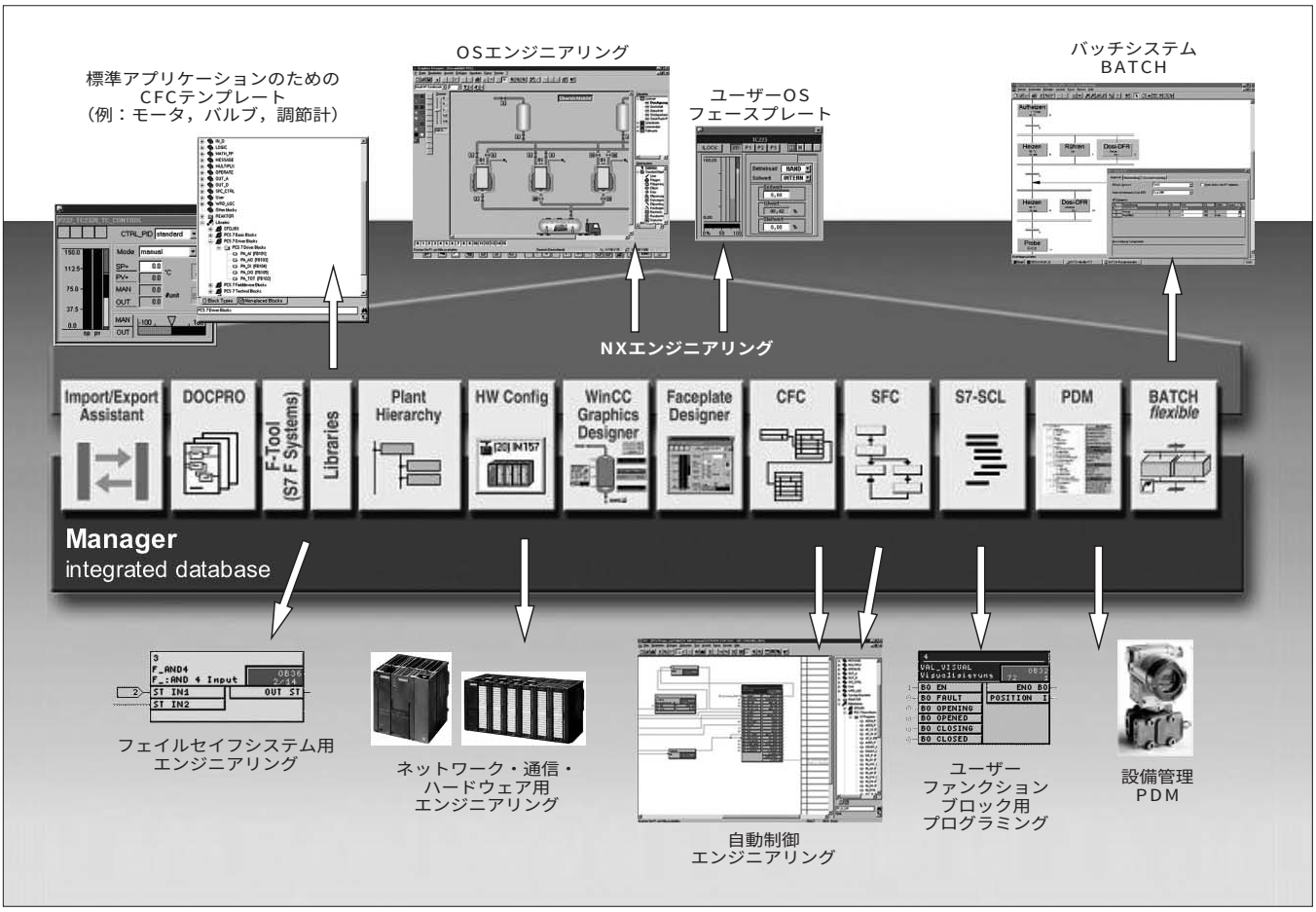
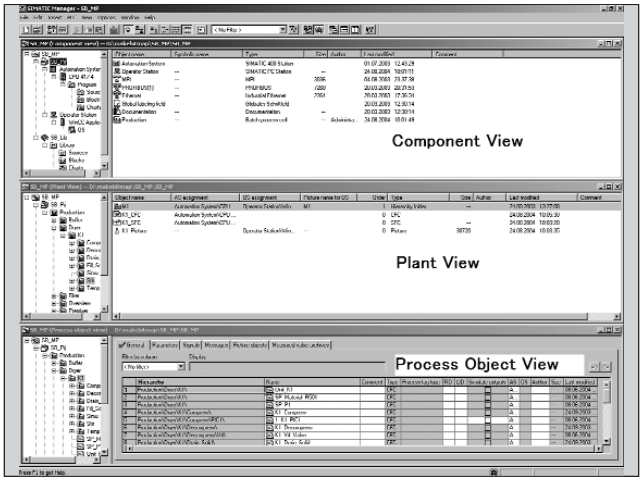


図２　View の構成



インテリジェントフィールド機器の集中的な設定，故障診断，ループチェックを行うことができる。

②　Plant View

制御対象となるプラントの分かりやすいオーバービューを目的とした階層管理アーキテクチャである。バッチ制御の国際標準である IEC61512（ISA S88）に対応した階層構造であり，Windows のエクスプローラライクな画面と操作で，実際のプラント階層構造に合わせた階層を作成することにより，コントローラ制御プログラムと HMI 画面を効率よく管理することができる。また，ここで定義されたプラント階層は OS の画面階層に自動展開される。

③　Process Object View

Plant View における階層管理では，散在して全体が把握し難い Process Object を集中管理する環境を提供する。個別の制御プログラムや HMI 画面に散在するバルブやモータなどの Process Object の各種パラメータを，個別ファイルを開くことなく，Excel〈注2〉ライクな画面と操作で一括した設定や変更ができる。Office〈注3〉との連携により，顧客から受領した I/O リストなどの各種リストからのコピーアンドペーストも容易である。

(4)　ライブラリ

MICREX-NX ではプロジェクトごとにライブラリを持ち，エンジニアは作成した制御プログラムをコピーアンドペーストの要領で容易にライブラリ化することができる。したがって，今まで培ってきた制御ノウハウを蓄積して，その後のエンジニアリングで活用していくことが容易である。また，折れ線近似などの機能ブロックや，制御ブロックとブロックアイコンおよびフェースプレートがセットに

〈注２〉Excel：米国 Microsoft Corp. の登録商標

〈注３〉Office：米国 Microsoft Corp. の登録商標

なったテクノロジカルブロックを多数搭載したライブラリパッケージも用意され，エンジニアは高度な制御を容易に実現することができる。

3　エンジニアリング機能と特徴

3.1　MICREX-NX のエンジニアリングの概要

MICREX-NX のエンジニアリングフローを**図3**に示す。MICREX-NX ではコントローラのエンジニアリングと HMI のエンジニアリングを行うシステムが相互連携し，効率的なエンジニアリングができる。

3.2　コントローラエンジニアリング機能と特徴

コントローラ制御プログラムの作成は，以下に述べる3種の言語を目的に合わせ，組み合わせて作成する。いずれの言語で作成しても，各制御プログラムにおいて使用される変数などのメモリは，そのプログラム内のみで使用できるように自動的に割り当てられるため，複数の制御プログラムから同一の物理メモリに書込みを行う二重書きの問題からエンジニアを解放するうえ，作成した制御プログラムのライブラリ化を容易にする。従来機種と比較しても機能が大幅に強化されているポイントである。

また，制御プログラムのデバッグは，ES に AS を接続するか，または ES 上でコントローラシミュレータである PLCSIM を動作させた状態で，各制御プログラムをオンラインモニタすることでで き，制御プログラムのデバッグ効率を大幅に向上することができる。

(1)　CFC（Continuous Function Chart）

IEC61131（422 ページの「解説」参照）の FBD に準拠する CFC は，MICREX-NX における連続制御プログラムの作成を行う。**図4**に CFC Editor の画面例を示す。本例は，モータの運転制御の例を示し，保護インタロック機能，操作機能，制御機能から構成される。CFC Editor 上では FB（Function Block），FC（Function Call）と呼ばれる制御ブロックをライブラリからドラッグアンドドロップし制御ブロック間を結線するだけで制御プログラムの作成ができる。制御ブロックとして，計装ループ制御，シーケンシャル制御，数値演算，論理演算など多彩なブロックが用意され，これらのプログラム構成要素を制御目的に応じて同一チャート上で自由に組み合わせて作成することができるため，富士電機の従来機種に見られた制御機能ごとの複雑な縦割りエンジニアリングから洗練されたフラットなエンジニアリングへと進化している。また，演算順序のカスタマイズや，制御ブロックの不使用端子の非表示など，きめ細かいニーズにも対応している。さらに，機能的にまとまったチャートを制御ブロック化することもでき，制御プログラムを容易に再利用できる環境を提供する。以上の特徴が有機的に相互作用し，エンジニアリング効率を向上させることができる。

(2)　SFC（Sequential Function Chart）

IEC61131 の SFC に準拠した，シーケンスによる逐次制御プログラムの作成を行う。**図5**に SFC Editor の画面例を示す。Step や Transition の作成における信号の割付は，制御ブロックのパラメータ一覧リストから必要な信号の選択を行うだけでできる。また，SFC Visualization パッ

図3　MICREX-NX のエンジニアリングフロー

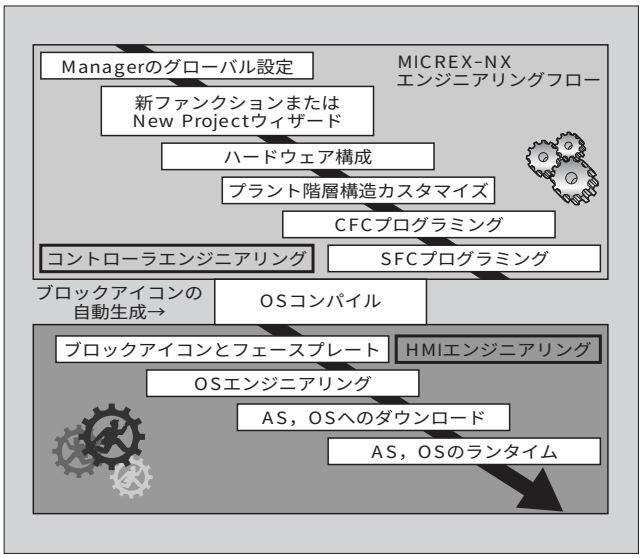


図4　CFC Editor の画面例

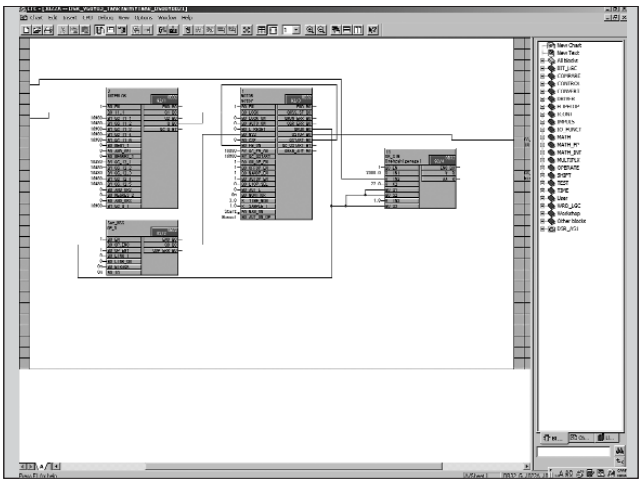
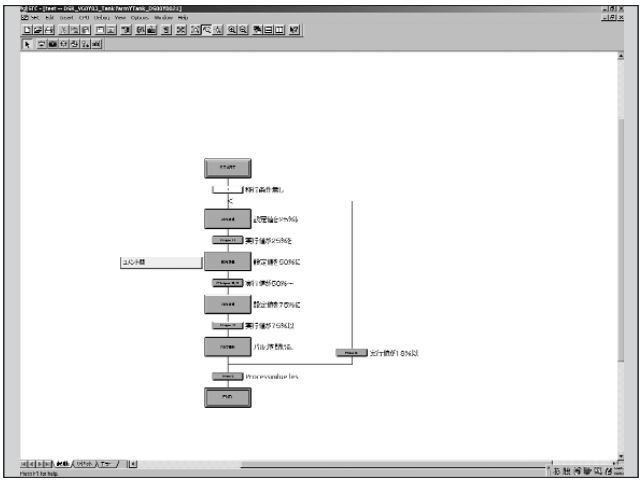


図5　SFC Editor の画面例



ケージを使用することにより，特別なエンジニアリングなしで，OS からオンラインで SFC の状態表示や操作ができる。

(3)　SCL（Structured Control Language）

IEC61131 の ST（Structured Text）に準拠するプログラミング言語である。プログラムに制御構造を持たせることができるため，主に CFC における FB/FC の作成に使用される。ユーザーが必要とする機能を自ら作成し，利用できるため，SCL を活用することでより自由度の高い制御プログラムの作成ができる。

3.3　HMI エンジニアリング機能と特徴

MICREX-NX における HMI エンジニアリングは，すでに述べたとおり，コントローラにおける制御プログラムの作成と密接に関連している。コントローラにおいて，HMI による表示や操作が前提の制御ブロック，例えば PID 制御などが作成された場合，HMI 画面上のシンボルとなるブロックアイコンと，ブロックアイコンをクリックすることで呼び出され，詳細な状態表示や操作・設定を行うウィンドウ，例えば PID 制御における計器図ウィンドウであるフェースプレートがあらかじめ指定した HMI 画面に自動生成される。一方，グラフィックデザイナにより HMI 画面をプロセスイメージに合わせて作成し，この上に自動生成されたブロックアイコンをイメージに合わせて再配置することで，HMI 画面が完成する。なお，フェースプレートは，Faceplate Designer を用いてユーザーニーズに合わせて作成することもできる。

(1)　WinCC Explorer

WinCC Explorer は，HMI エンジニアリングの共通プラットフォームである。上述のグラフィックデザイナもその一つであり，ほかにも，グローバルスクリプトやユーザー管理など多様な機能を提供する。

(2)　グラフィックデザイナ

図 6 にグラフィックデザイナの画面例を示す。HMI 画面の作成ソフトウェアであるグラフィックデザイナは，汎用の画像処理ソフトウェアと同様の操作性と表現力で HMI 画面の作成を支援する。また，ユーザーがライブラリを作ることができるため，過去に作った HMI の部品を蓄積して再利用できるほか，画面操作性の向上など画面作成の効率向上につながるような仕組みを提供している。

(3)　グローバルスクリプト

C 言語または VB（Visual Basic）言語に準拠したスクリプトを実装し，汎用 SCADA に相当する柔軟な画面のカスタマイズができる。

(4)　ユーザー管理

プラントのセキュリティやトレーサビリティの見地からすべてのオペレーターへのきめ細かな権限の付与を行う。操作の許可といったオペレーションレベルを，設備単位で指定することができる。

3.4　支援ソフトウェア

MICREX-NX エンジニアリングの効率および品質を向上させるツール群を紹介する。

(1)　Import Export Assistant

同じ機能を持つ複数の制御プログラムを一括で作成する機能を提供する。制御プログラムのひな型を作成し，作成先や入出力パラメータなどの制御プログラムごとに異なる情報を Excel ライクな画面と操作で一括設定し，複数プログラムを展開することができる。展開を行った設定ファイルは ES に保存できるため，制御プログラムに変更が必要な場合もすでにある設定ファイルを用いて即座に複数の制御プログラムに変更を反映することができる。**図 7** に Import Export Assistant の概要を示す。

(2)　Version Cross Checker

作成した制御プログラムのバージョン管理ができる。このソフトウェアは，変更前後のプロジェクトについてその追加，削除および変更箇所と内容を反転表示によりグラフィカルに確認できる機能を備えているため，制御プログラムの版数管理，障害解析など幅広い用途で用いることができる。**図 8** に Version Cross Checker による比較結果の画面例を示す。

(3)　コントローラシミュレータ（PLCSIM）

MICREX-NX ではコントローラのシミュレーションを

図 6　グラフィックデザイナの画面例

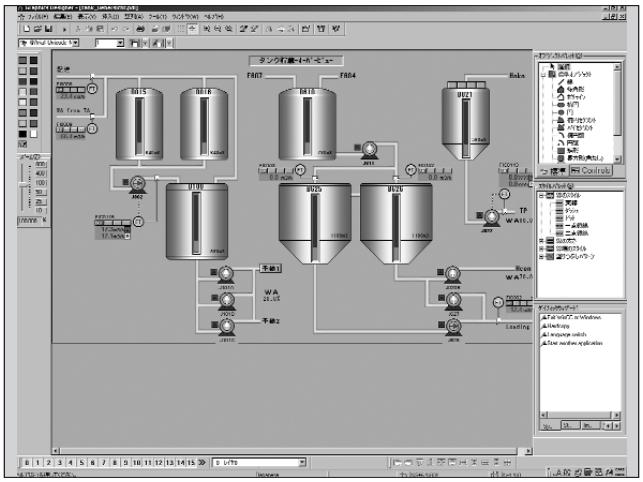


図 7　Import Export Assistant の概要

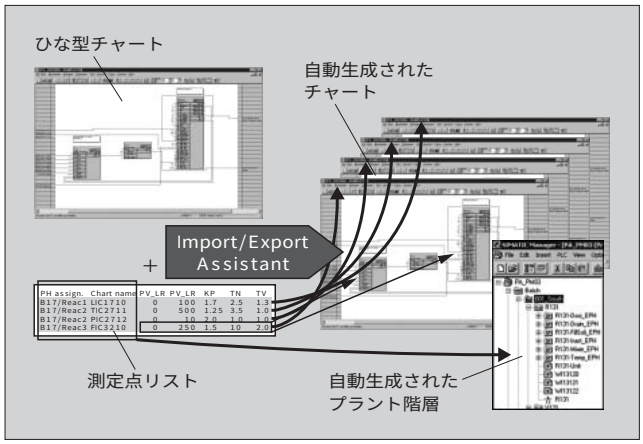
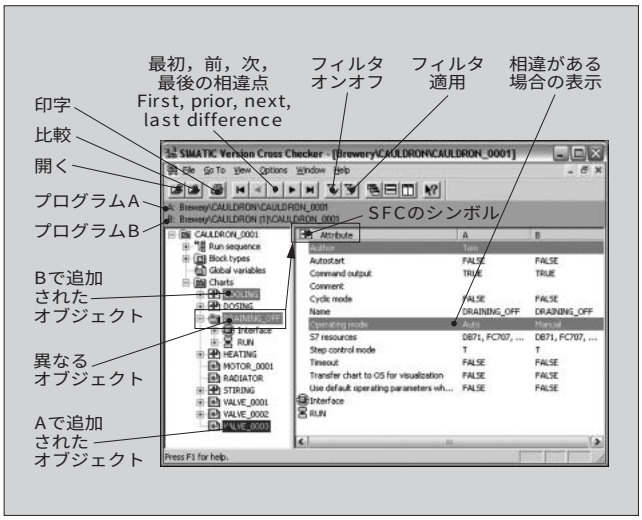


図8　Version Cross Checkerの画面例



ESパソコン上で実行することができる。シミュレーション環境は容易に構築でき，1台のESパソコンでOS，ASが各1台接続されているのと同等のデバッグができる。これにより，コントローラとHMI画面のデバッグ効率が大幅に向上する。また1台のパソコンへのオールインワンにより可搬性が向上し，出先での機能確認など場所を選ばない対応ができる。

4　あとがき

洗練された機能を持つMICREX-NXエンジニアリング支援ツールにより，これまでに培ってきた制御のノウハウを十分に活用し市場の要求に応えられる高効率・高品質なエンジニアリングを実現した。今後，さらに理想的なエンジニアリングプラットフォームを提供し，プロセスオートメーションシステムの発展に寄与していく所存である。

解　説　IEC61131

IEC（International Electrotechnical Commission）では，1979年に作業部会を設置し，プログラマブルコントローラのハードウェア設計，インストール，テスト，ドキュメンテーション，プログラミング，通信からなるプログラマブルコントローラの仕様の標準化を行った。仕様は以下のとおり，五つのPartに分かれる。

Part1：General information（一般情報）
Part2：Equipment and test requirements（機器要求事項および試験）
Part3：Programming languages（プログラム言語）
Part4：User guidelines（使用者の手引）
Part5：Communications（通信）

このうち，Part3がプログラム言語に関する規定であり，次の5言語が規定されている。

(1)　SFC（Sequential Function Chart）
(2)　IL（Instruction List）
(3)　ST（Structured Text）
(4)　LD（Ladder Diagram）
(5)　FBD（Function Block Diagram）

MICREX-NXには，SFC，STおよびFBDに準拠したプログラミング言語があり，ユーザーは制御内容に応じて，適した言語を使い分けることができる。

e-Front runners

＊本誌に記載されている会社名および製品名は，それぞれの会社が所有する
商標または登録商標である場合があります。