

# S88 入門

バッチシステムをよりよくデザインするために



# 著者・編集者リスト

メンバー 敬称略 五十音順

伊豆 和巳	呉羽テクノエンジ株式会社
伊藤 利昭	名古屋工業大学
岩根 伸久	三井化学株式会社
桑谷 資一	横河電機株式会社
坂井 幹	株式会社山武
篠田 拓也	横河電機株式会社
杉浦 彰俊	森永乳業株式会社
高山 仁	株式会社山武
塚本 泰弘	三井化学株式会社
西本 吉孝	三井化学株式会社
野里 一七	株式会社日立ハイテクノロジーズ
藤咲 範男	株式会社日立ハイテクノロジーズ
淵野 哲郎	東京工業大学
星 宏一	呉羽テクノエンジ株式会社
三上 恭男	株式会社山武
脇山 昇	横河電機株式会社

# 目次

まえがき	3
<b>1 製造現場の抱える悩み</b>	<b>5</b>
1.1 バッチプラントの概要およびバッチシステム設計上の問題点( アンケート結果 )	6
1.2 バッチシステムを増改造する際の悩み	6
1.3 既設設備を基に、バッチシステムを設計する際の悩み	7
1.4 開発研究を基に、バッチシステムを設計する際の悩み	7
1.5 開発期間の短縮、グローバル展開等の課題	8
<b>2 S88標準化によるバッチシステム設計</b>	<b>9</b>
2.1 S88で扱うモデル	10
2.1.1 プロセスモデル( Process Model )	10
2.1.2 物理モデル( Physical Model )	11
2.1.3 手順制御モデル( Procedural Control Model )	13
2.2 処方手順と設備の実体とのマッピング	15
2.3 S88標準化手法におけるポイント	16
2.4 設計作業の流れ	18
<b>3 機能仕様書例</b>	<b>19</b>
おわりに	29
<b>付録A バッチ制御 - 第1部:モデル及び用語</b>	
解説(日本規格協会発行 JIS C 1807 バッチ制御- 第1部:モデル及び用語(2002)からの抜粋)	32
1. 制定の経緯及び趣旨	32
2. 四つのモデルとモデル間の独立性	33
3. 4種類の処方とその対応分野	34
<b>付録B 寄稿 『製品開発とレシピエンジニアリング』 名古屋工業大学 伊藤 利昭</b>	<b>36</b>
1. 製品寿命の短命化とエンジニアリング業務の変化	36
2. レシピエンジニアリングとは	37
3. レシピエンジニアリングに関する技術課題と研究・開発状況	39
4. 産業の枠を越えたレシピエンジニアリング	40



# まえがき

## S88にあまり馴染みのない皆様へ

バッチシステムをよりよくデザインするためにS88が提唱する標準化が注目されてきています。しかしながら一般に『S88とは何か?』、『どのようなメリットがあるのか?』と言うS88に関する基本的な理解が進んでいません。そこでS88を分かり易く紹介し、出来るだけ多くの方々に少しでもS88について理解を深めていただきたいと思い本書を作りました。

## 読んで欲しい読者

もしあなたがバッチプロセスの生産にちょっとでも関わり合っていてバッチシステムの構築、運転、保守に一度でも携わったことがあり、そしてそこで少なからず苦い経験をされた方は、是非この本を読んでみてください。

もちろんこれからバッチシステムについて勉強を始めようと考えられている方も読んでみてください。

## S88とは何か?

S88はバッチに関する国際標準規格の総称です。

もともとはISAの標準化WGが始めた活動で1988年に活動を開始しました。

彼等が活動を始めた動機はバッチ制御に関する悩みを解決するためでした。

- バッチ制御の雛型(何処にでも、誰にでも通用する)が無い。
- バッチプロセスでやりたいことを業者に伝えるのが大変(通じない)。
- バッチ制御システムを作り上げる労力が大変(手間がかかる)。
- 複数のメーカーの制御システムを一つにまとめるのは、エンジニアにとって苦痛(地獄)。

S88はこんな課題を解決するために提案され、その後国際標準として制定され、JISにもなっています。

S88は導入メリットがたくさんあります。

第1章で述べている製造現場が抱える悩みの全てを解決できる訳ではありませんが、以下の解決のヒントがS88にあります。

- 標準化、再利用によるコスト削減、リードタイム短縮が図れる。(特に増設改造時)
- 技術の伝承が容易。
- 複数メーカーの制御システム導入の困難さが軽減される。

はっきり申し上げてS88はかなり判りづらいです。判りづらさの原因の1つにS88用語があります。本書では、できるだけ平易な表現や具体例を使用するように心がけました。

S88の考え方をよく理解してもらうために、リンゴジュース製造プロセスを例に説明を進めます。本書に登場する製造プロセスは架空のものです。実際の食品プロセスとは全く異なったプロセスになっていますが、あくまでS88のモデルアプローチを分かり易く説明するための題材ですのでご注意ください。

本書をお読みになって、少しでもこれは良いなと感じられたところを実務に役立てていただければ幸いです。



# 第1章

## 1 製造現場の抱える悩み

1.1 バッチプラントの概要および バッチシステム設計上の問題点( アンケート結果 )	6
1.2 バッチシステムを増改造する際の悩み	6
1.3 既設設備を基に、バッチシステムを設計する際の悩み	7
1.4 開発研究を基に、バッチシステムを設計する際の悩み	7
1.5 開発期間の短縮、グローバル展開等の課題	8

# 1 製造現場の抱える悩み

## 1.1 バッチプラントの概要およびバッチシステム設計上の問題点(アンケート結果)

日本学術振興会プロセスシステム工学第143委員会(PSE143)ワークショップNo.17が実施したアンケートより、「バッチプラントの概要」として以下の回答が寄せられました。

- バッチプラントによる生産高が半分以上である事業所が50%弱を占める。
- シングルパスのバッチプラントは全体の45%であり、残りはマルチパスやネットワーク型の複雑な構造である。
- 単一製品のプラントは全体の約30%で、残りは複数製品を製造している。1製品当たりの品種数はほとんどの業種で100品種以下であるが、3,000品種に達する場合もある。そして、製品または品種切り替えは85%のプラントで行われている。
- バッチサイクルは1日未満が73%であるが、食品、医薬では3週間に及ぶ例もある。ほとんどの場合は計器室中心または現場と計器室の協調による運転操作である。

また、「バッチシステム設計上の問題点」として「既存システムの変更や改造が大変である」「設計手法の標準化が遅れている」など多数の項目があげられていました。そこで、ケース別に「製造現場の抱える悩み」を紹介します。

## 1.2 バッチシステムを増改造する際の悩み

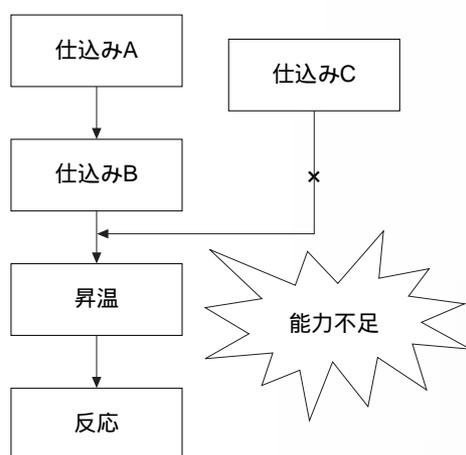


図 1.1 単位シーケンス追加時の悩み

1つの工程管理シーケンスで多品種を製造する場合、すべての品種に対応できるように単位シーケンスを並べ、使用しない工程はパスするといった形式で設計されるケースが多く見受けられます。導入時の制御システム設計者がDCS能力をフルに使用した場合、単位シーケンスを追加しようとしても、能力不足のため対応できず、手動操作で対応せざるを得ないケースがあります(図 1.1)。

仮に、品種毎に必要なシーケンスが再利用できる単位で実装されていれば、工程順序の入れ替えを問題なく行えると同時に増改造が容易になります。さらに、単位シーケンスを同時(並行)に動かすことができれば「設備の回転率」を高めることができ、生産性が飛躍的に向上することになります。

また、DCSのアップグレードにおいて、同一メーカーであっても機種毎にプログラム言語が異なり、言語を全て人が書き換えなければならないケースが散見されます。

例えば、プログラムはテキストファイルで作成し、必要な言語に自動変換できるようなシステムにはならないのでしょうか。

### 1.3 既設設備を基に、バッチシステムを設計する際の悩み

バッチプラントでは製品の移り変わりが激しく、多くの場合、その都度プロセスの改造を要求されます。合理的に改造を行うためには、既設設備を有効利用すべきですが、ベンダーによって制御システム仕様書が異なる、既存のアプリケーションが流用できない等の理由により、設備の更新を余儀なくされるケースが見受けられます。

制御システム仕様書が統一されると、設備の再利用率も高まり、更に、既設設備+ の改善で、容易かつ短期間でのコピープラント建設が可能となるでしょう。

更に、バッチプロセスの設計思想が個人に帰属することも防止できるため、設計担当者が配置転換等でいなくなった場合でも、大きな問題は生じないと考えられます。技術伝承という観点からも、制御システム仕様書の統一が望まれています。

### 1.4 開発研究を基に、バッチシステムを設計する際の悩み

研究段階(フラスコレベル)での操作は主に手動で行われ、複数のフラスコを同時に操作することもほとんど有りません。「品質」が最重視され「安全」「コスト」が軽視されるケースが見受けられます。一方、実装段階では「複数設備を安全かつ効率的に操作するにはどうすべきか」がポイントとなります。スケールアップに伴うパラメータの変更や、原材料の制約、法規制等、設備特有の情報が付加されても(図1.2)、研究段階と同じ品質の製品を安全かつ経済的に得ることができる理想的なプロセス設計手法はないのでしょうか。

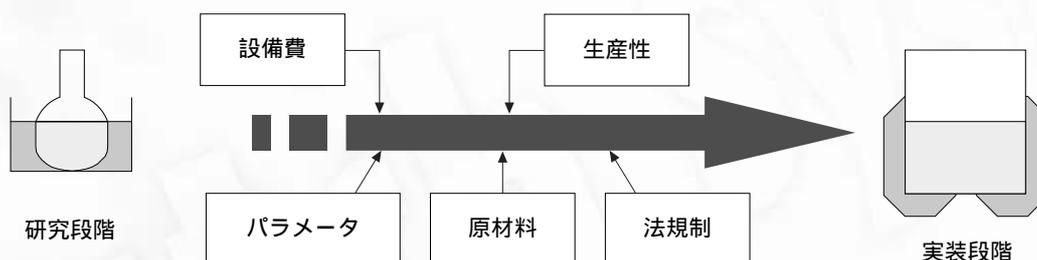


図1.2 研究段階から実装段階に至るまでの流れ

### 1.5 開発期間の短縮、グローバル展開等の課題

バッチプロセスシステムを設計する上で最も工数を要する作業は「ユーザーとベンダー / エンジニアリング間」あるいは「研究開発と設計 / 運転間」での情報交換です。これらは開発期間の長期化、結果としてコストとして跳ね返ります。近年の製品ライフサイクルを考えると開発期間はできるだけ短くしなければなりません。例えば、メーカーが海外進出を考えた場合、システムの仕様が海外と異なると、一からシステム開発を行わなければならない、場合によっては機会を逸してしまう危険性があります。

このようなリスクを回避し実装化へのスピードアップを図るためには、誰にでも容易に理解できる「世界標準」のプロセス設計手法の確立が望まれます。

# 第2章

## 2 S88標準化によるバッチシステム設計

2.1 S88で扱うモデル 10

---

2.2 処方手順と設備の実体とのマッピング 15

---

2.3 S88標準化手法におけるポイント 16

---

2.4 設計作業の流れ 18

---

## 2 S88標準化によるバッチシステム設計

S88標準化によるバッチシステム設計の狙いのひとつに、モデルの概念の採用によりだれでも同じ体系でバッチシステムを設計できることがあります。

もうひとつの狙いとして製品が変わるたびに制御システムのソフトウェアの改造が発生してしまうという問題を解決することがあげられます。S88では、“製品に依存している製造の手順”と“設備に依存している制御”を分けることによってこの問題の解決を図っています。詳しくは、“2.2 処方手順と設備の実体とのマッピング”で説明します。

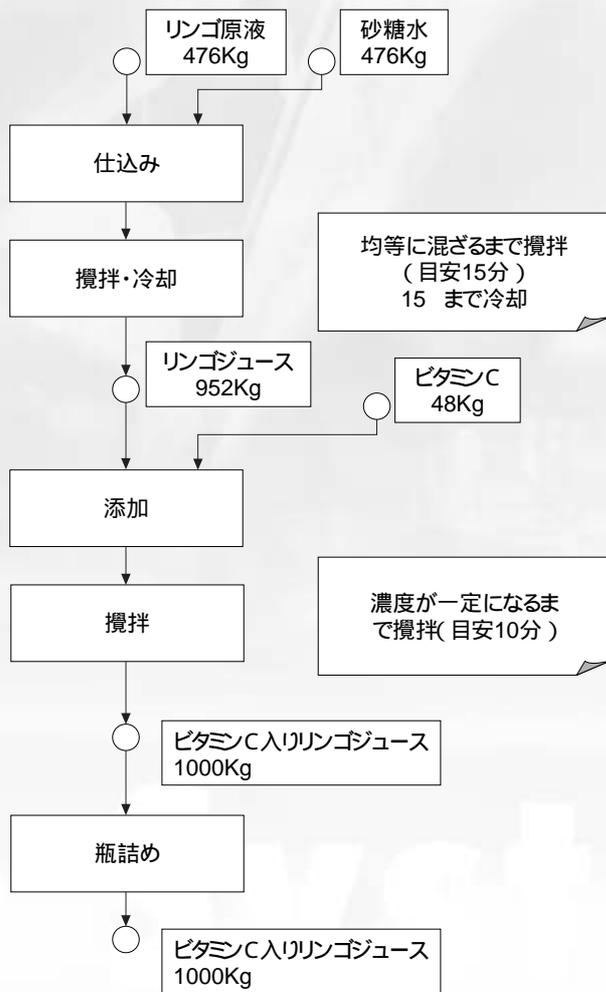
### 2.1 S88で扱うモデル

特に重要な3つのモデル、

- プロセスモデル( Process Model ): 化学反応や物理反応などのプロセス面から見た階層化
  - 物理モデル( Physical Model ): ハードウェアの面から見た階層化
  - 手順制御モデル( Procedural Control Model ): ソフトウェアの面から見た階層化
- について説明します。

#### 2.1.1 プロセスモデル( Process Model )

プロセスモデルとは、“ある物質を作るために、何を行うか”を書いたものです。通常ものを作る場合には、どういう原料をどのような割合で使用するのか、触媒を使用するのか、温度は何度まで上げるのか等といった化学的、物理的变化を記していきます。これがプロセスモデルです。ビタミンC入りリンゴジュースを例にとると、図2.1のようなプロセスモデルになります。製品になるまでの物質の化学的、物理的变化という点のみ記述したものであるため、設備の概念を持ちません。“送液”など設備によって変わる動作表現は排され、実験室レベルから商業プラントまでプロセスモデルは同じ表現となります。



原単位表  
1000kgベース

原料	量	単位
リンゴ原液	476	Kg
砂糖水	476	Kg
ビタミンC	48	Kg

図 2.1 ビタミン C 入りリンゴジュースのプロセスモデル (Process Model)

### 2.1.2 物理モデル (Physical Model)

プロセスモデルができあがると、このプロセスモデルを元に実際にビタミンC入りリンゴジュースを製造するプラントを建設することになります。新しいプラントを建設するのではなく、既存のプラントでビタミンC入りリンゴジュースを製造する場合がありますが、ここでは省略します。

リンゴ原液、砂糖水、ビタミンCをひとつの調合槽で混合することも可能ですが、ここではリンゴ原液と砂糖水を混ぜてリンゴジュースを作り、次にリンゴジュースにビタミンCを混ぜてビタミンC入りリンゴジュースを作ることになります(図 2.2)。瓶詰め工程は割愛します。

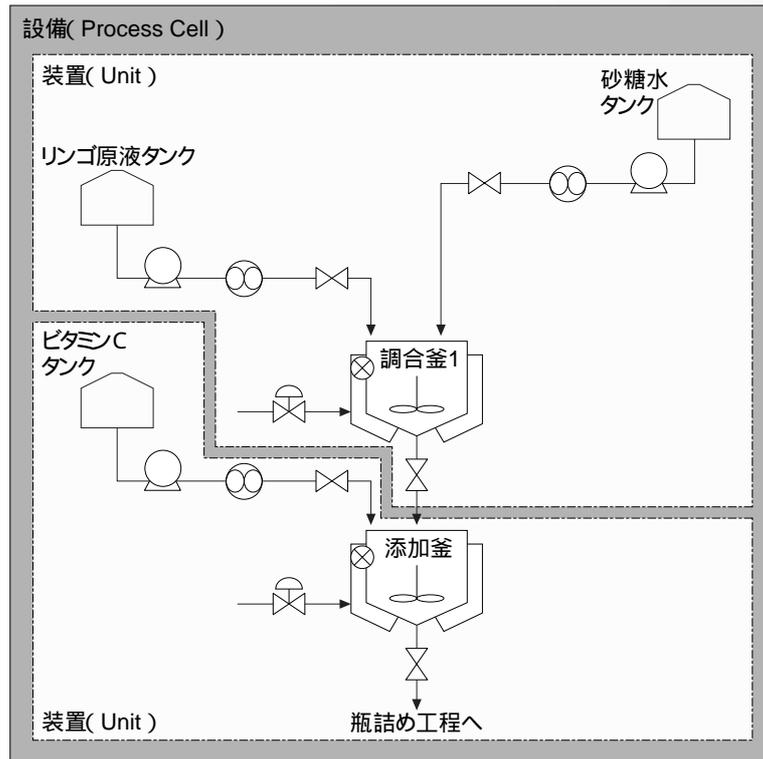


図 2.2 リンゴジュース製造プラント

物理モデル (Physical Model) では、プラントを階層分けしています。プラント全体は設備 (Process Cell) と呼ばれ、“設備” を分解してゆくと、装置 (Unit)、次に機器 (Equipment Module) と分解されます (図 2.3、図 2.4)。なお機器 (Equipment Module) とは仕込み、攪拌、温調といったひとつの機能を司るものであり、バルブ、ポンプ等の計装機器 (Control Module) や配管などの集まりのことです。

バッチ制御でのソフトウェアの汎用化、再利用性を考慮すると、機器 (Equipment Module) のくくり方が鍵となります。

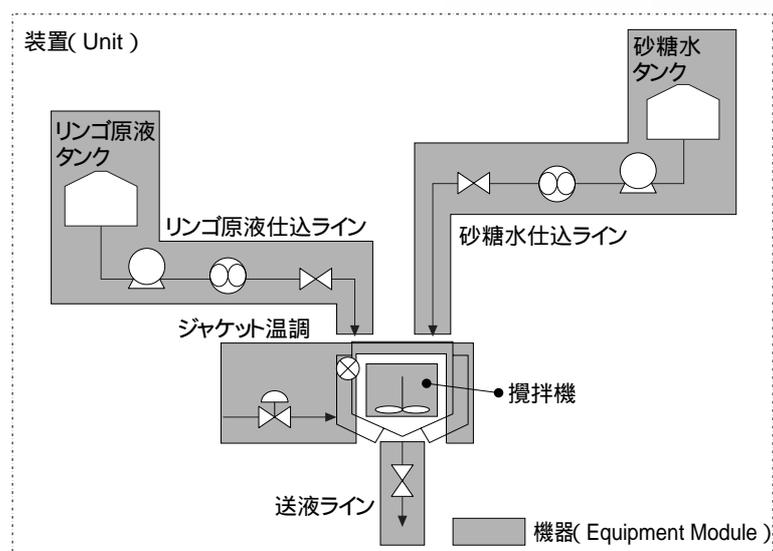


図 2.3 装置 (Unit) の分割

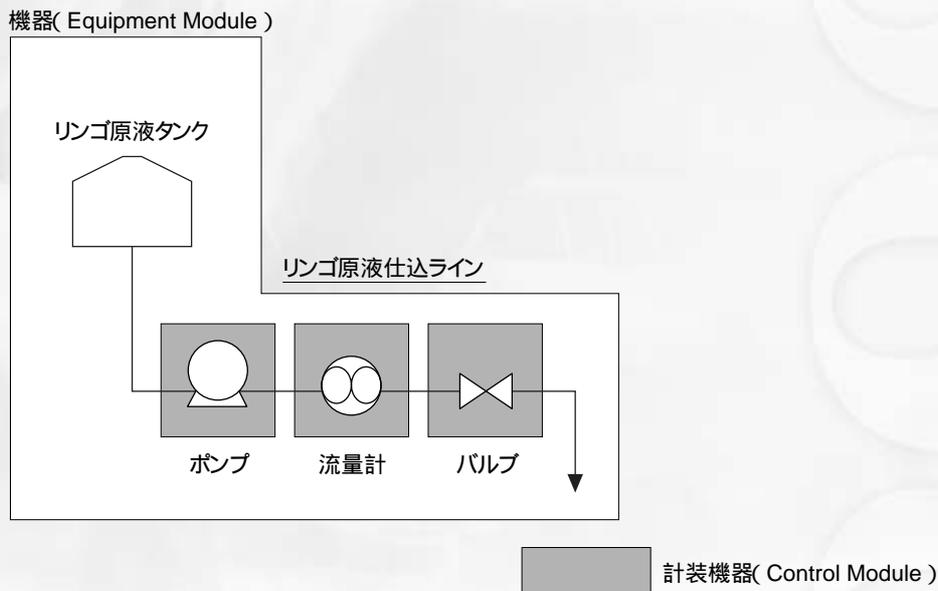


図 2.4 機器( Equipment Module )の分割

### 2.1.3 手順制御モデル( Procedural Control Model )

物理モデルができあがると、プロセスモデル及び物理モデルから、リンゴジュース製造の手順は以下となります。

- |                                 |       |             |
|---------------------------------|-------|-------------|
| 1. リンゴ原液を476.0Kg仕込む . . . . .   | } 仕込み | } リンゴジュース調合 |
| 2. 砂糖水を 476.0Kg仕込む . . . . .    |       |             |
| 3. 15分間攪拌する . . . . .           |       |             |
| 4. 15 まで冷却する . . . . .          | } 冷却  |             |
| 5. 送液する . . . . .               | } 送液  |             |
| 6. ビタミンC溶液を 48.0Kg仕込む . . . . . | } 添加  | } ビタミンC添加   |
| 7. 10分間攪拌する . . . . .           | } 送液  |             |
| 8. スtockタンクに送液する . . . . .      |       |             |

ビタミンC入りリンゴジュースは大きく2つの単位手順( Unit Procedure )に分かれます。リンゴ原液と砂糖水からリンゴジュースを調合する単位手順、ビタミンCを添加する単位手順の2つです。リンゴジュースを調合する単位手順を細かく分けると原料を仕込む、冷却する、送液するという3つの単位操作( Operation )に分かれます。原料を仕込む単位操作はリンゴ原液仕込み、砂糖水仕込み、攪拌という3つのフェーズ( Phase )に分かれます。このように手順は大きなくくりから小さなくりに階層化することができます。図 2.5 はリンゴジュース製造の手順を PFQ( Procedural Function Chart ) を使用し表記した例です。このような手順の階層化したものを手順制御モデル( Procedural Control Model ) と呼びます。

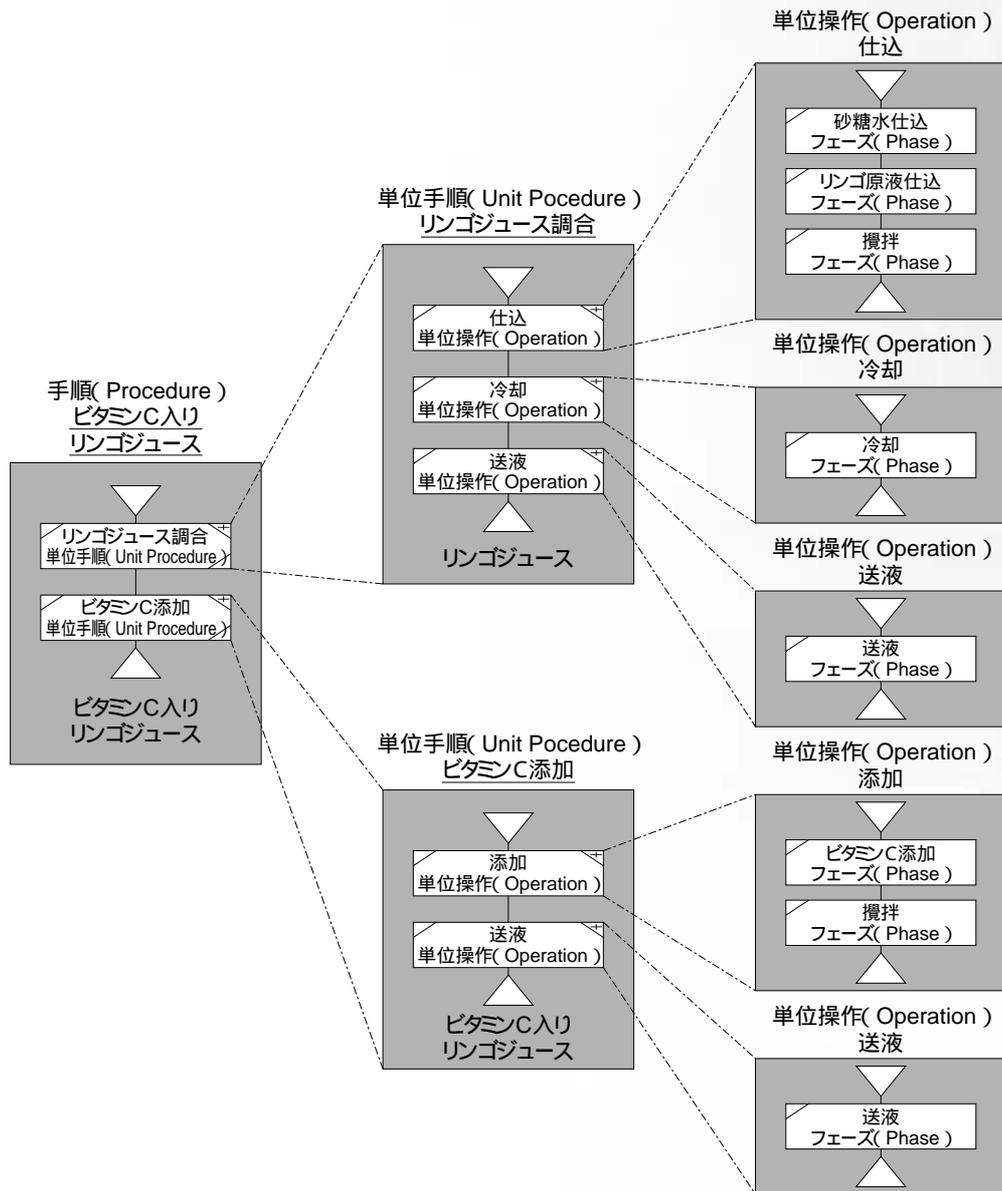


図 2.5 リンゴジュース製造の手順を PFC で表記した例

プロセスモデル、物理モデル、手順制御モデルは密接に関係しており、どれかひとつが変われば他にも影響を受ける関係になっています。例えば、上述の物理モデルの説明では、調合槽と添加槽の2つでプラントを構成しましたが、調合槽のみでビタミンC入りリンゴジュースを作ることも可能です。その場合は、調合槽から添加槽への“送液”が不要となるため、手順制御モデルが変わってきます。また逆にバッチタイム短縮などのために手順を簡略化したい場合には物理モデルを変更(プラントを変更)する場合があります。

S88標準化によりバッチシステム設計を行なう場合はこれらのモデルを意識し、プロセスモデルより物理モデルの設計(=プラントの設計)、手順制御モデルの設計(=制御手順の設計)を行います。相互に影響しあうため、適切な設計のためにはそれぞれの設計情報によりお互いの設計を軌道修正する必要があります。

## 2.2 処方手順と設備の実体とのマッピング

図 2.6 は S88 標準化による手順制御モデルと物理モデルの関係です。ここで設備制御 (Equipment Control) とは仕込み、攪拌、温調、送液といった設備固有の振る舞いのことです。

製造の手順は手順制御モデルで記述されます。これを処方手順 (Recipe Procedure) と呼び、このリンゴジュース製造プロセスの例では最下層のフェーズ (Phase) が仕込みや攪拌などの機器にマッピングされています。バッチ運転時には処方手順に従い自動的に制御システムが機器に指示を与え、製造が行なわれてゆきます。

このように設備制御を処方手順と切り離し制御システムに実装することにより、手順制御モデルを変更するだけで制御動作を変えることが可能です。コンタミネーション防止、最適資源配分のための資源割り当てについても同様に処方手順と切り離し実装することにより、手順制御モデルを変更するだけで資源割り当てを変えることが可能です。また製品毎に製造の手順が異なったとしても、使用する機器は同じであるため処方手順さえ変えれば良いわけです。

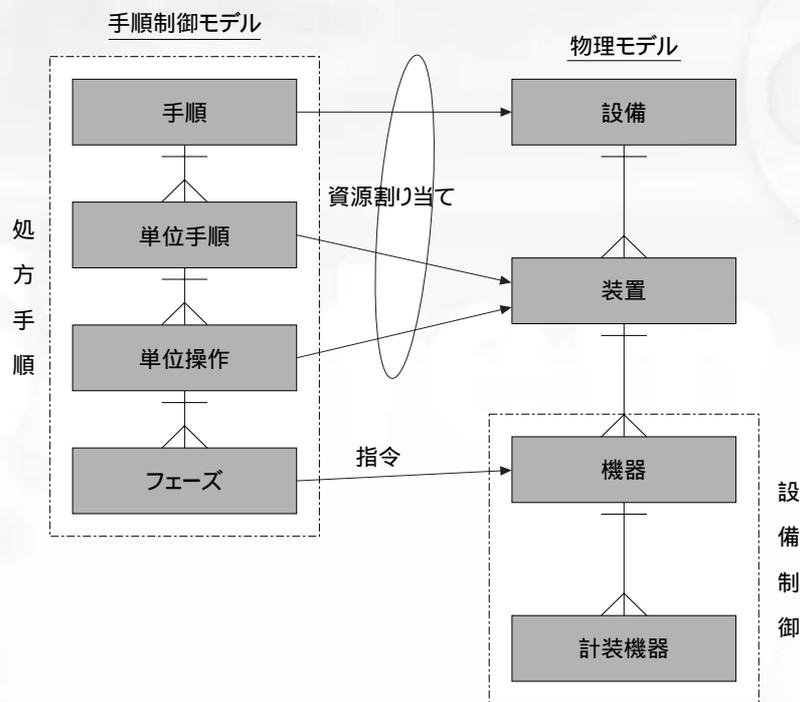


図 2.6 処方手順と設備の実体とのマッピング例

### 2.3 S88標準化手法におけるポイント

従来のバッチ制御では、手順は固定で設定値のみが可変(処方と呼ばれるものは一般的に銘柄毎の設定値のことを指していました)となっており、手順変更の際はソフトウェアの改造を伴うことが多かったのですが、S88では処方(Recipe)は銘柄毎の設定値ばかりでなく、図2.5のような製造の手順も含んだもの(=処方手順)と定義しており、ソフトウェアコストを軽減する考え方となっています。

工程時間短縮のために新しい製法に変更する場合を考えてみましょう。図2.5では砂糖水を仕込んだ後にリンゴ原液を順番に仕込んでいますが、これを時間短縮のため2つの原料を同時に仕込む製法に変えると図2.7となります。S88階層化構造では変更された箇所はリンゴジュース調合の仕込み工程のみであり、ソフトウェア変更の影響箇所が非常に限られている事がわかります。ビタミンC添加の後に別の添加剤を追加する場合や別のジュースを新たに試作する場合もこの手法であればソフトウェア変更が最小限で済みます。このようにモデル化の手法を使い製品と設備の制御の分離を整理し、制御システムへの実装のために製品(処方)と設備制御をマッピングさせる手法を用いることによりソフトウェア変更が容易となり柔軟な運用、素早い新製品試作、市場への投入が可能になります。

なお、物理モデルや手順制御モデルを図2.8のように設計してしまうと、砂糖水とリンゴ原液をシリーズに仕込むことができなくなります。フェーズ(Phase)、機器(Equipment Module)を適切な区分で設計することが重要です。

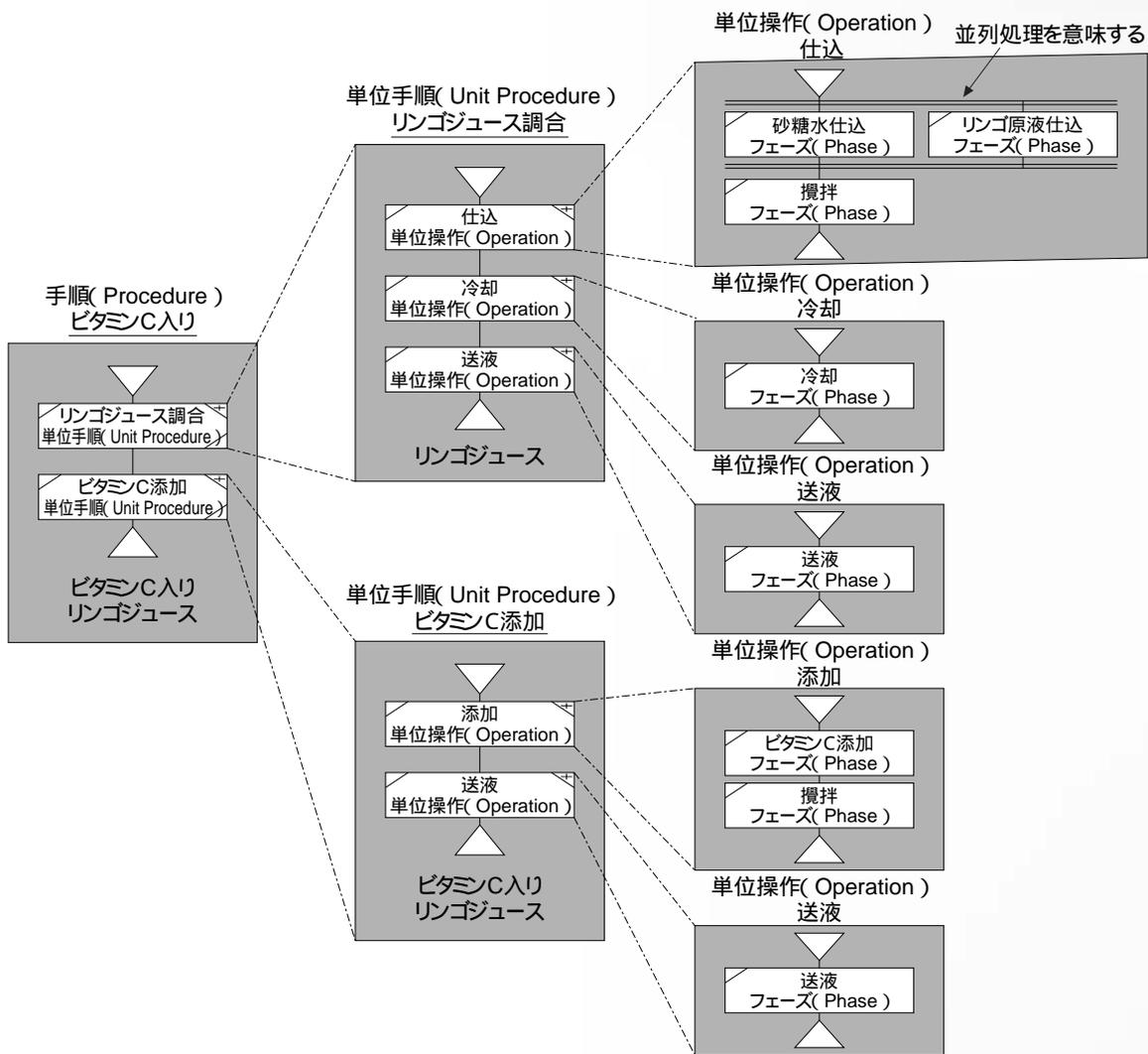


図 2.7 バッチタイム短縮後の処方手順(Recipe Procedure)

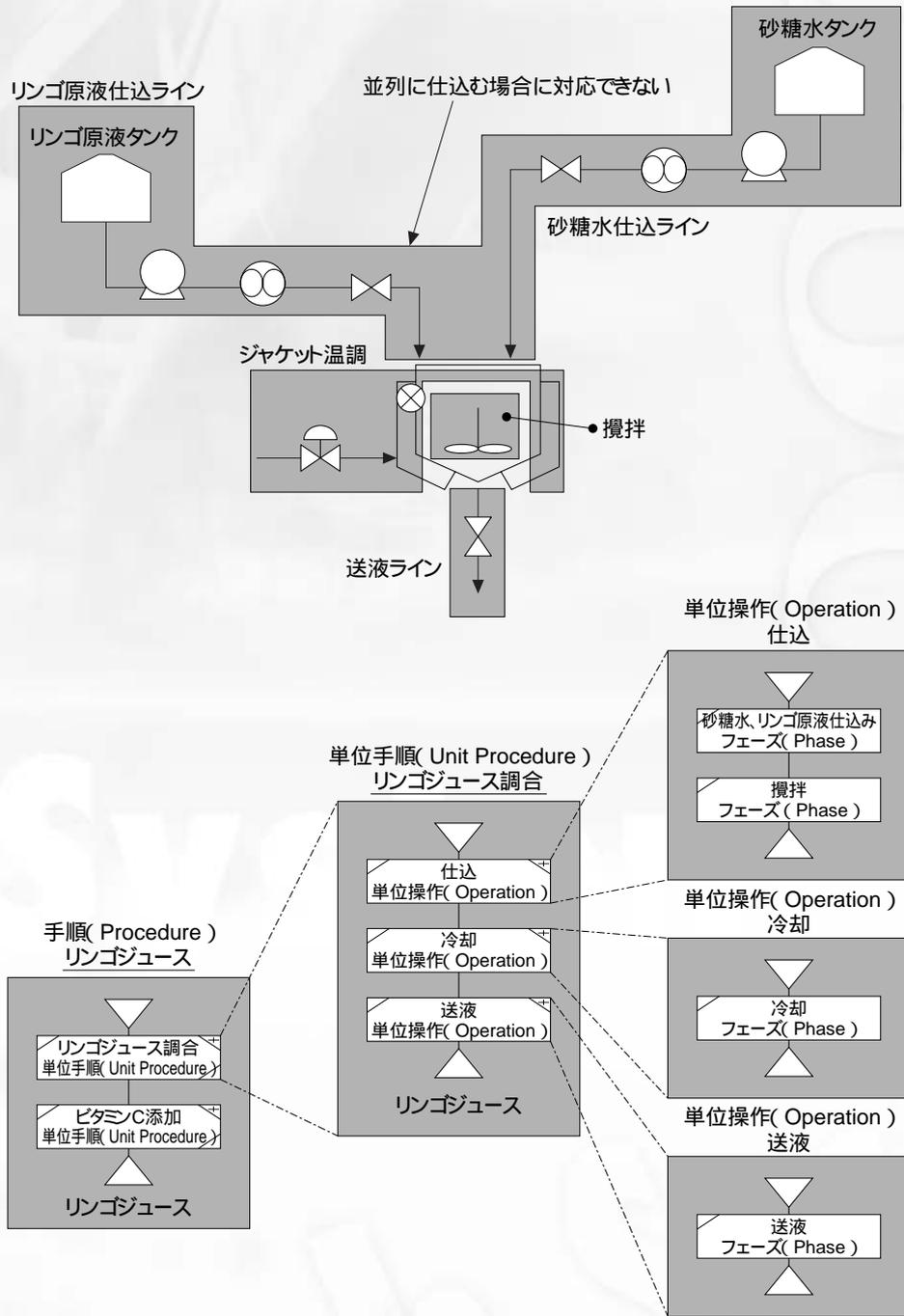


図 2.8 適切な区分で設計されていない機器 / 処方手順の例



# 第3章

## 3 機能仕様書例

System

Batch System

## 3 機能仕様書例

ここでは2章で紹介した標準化手法を使った仕様書の具体例を記載します。

仕様書の構成は大きく3つに分けています。

### ◆製品ベース仕様

ここでは、原料から製品となるまでの加工プロセスの視点から記述します。

具体的には、どんな原材料でどのような手順で加工していくかを説明します。

### ◆設備ベース仕様

ここでは、製品ベース仕様で述べた加工プロセスを具体的には、どんな設備で生産していくかを説明します。

### ◆共通仕様

ここでは、制御システムを構築する上で必須の共通項目を説明します。

例3.1に仕様書の目次例を示します。

## リンゴジュース生産設備制御システム仕様書

### < 目次 >

#### 1. 製品ベース仕様

1. プロセス
2. リンゴジュース製造プロセス
  1. ビタミンC入りリンゴジュース製造プロセス
3. 処方
  1. リンゴジュース製造処方
  2. ビタミンC入りリンゴジュース製造処方

#### 3. 共通仕様

1. オペレーション仕様
2. 異常処理の考え方
3. 共有資源
4. 経路管理
5. 工程間同期

#### 2. 設備ベース仕様

1. 設備関連図
  1. 設備階層図：Unit 1
  2. 設備階層図：Unit 2（省略）
  3. 設備階層図：Unit 3（省略）
  4. 設備階層図：Unit 4（省略）
2. 機器設計図
  1. EM1（砂糖水仕込み機器制御）
  2. EM2（ビタミンC仕込み機器制御）（省略）

REV \_\_\_\_\_ REMARKS \_\_\_\_\_ PJDocNo. \_\_\_\_\_ TITLE \_\_\_\_\_ BLOCK CODE \_\_\_\_\_

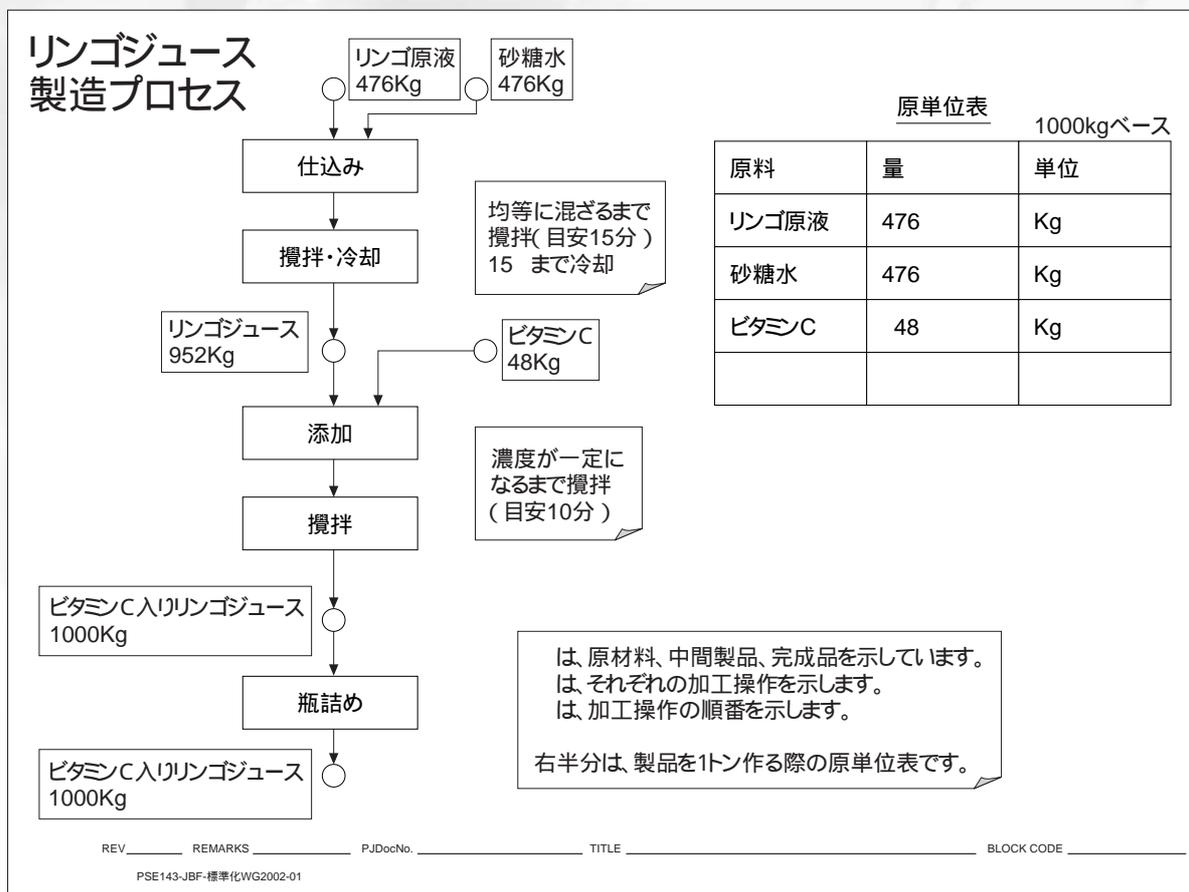
PSE143-JBF-標準化WG2002-01

### 例3.1 仕様書目次

## 製品ベース仕様

例 3.2 は、化学反応や物理反応などプロセス面から見た仕様書例です。

ある物質を作るために何を行うかを記述したものであり、製品になるまでの物質の化学的、物理的变化という点のみ記述したものであるため、設備の概念を持ちません。"送液"など設備によって変わる動作表現は排され、実験室レベルから商業プラントまでプロセスモデルは同じ表現となります。



例 3.2 リンゴジュース製造プロセス仕様

例 3.3 は、リンゴジュース製造の基本処方の例です。右半分は、手順制御モデルの例です。

## リンゴジュース製造処方

**【ヘッダ】** 商品名  $\times \times$  (開発コード R101)  
 特徴 従来品( R100 )にビタミンCを添加したタイプのリンゴジュース

改訂番号	日付	訂正内容	承認	担当
0	$\times \times$	初版発行	(印)	(印)

**【フォーミュラ】《原料》**

原料名	調合釜1( 10t )	調合釜2( 30t )
リンゴ原液	4,760 kg	14,280 kg
砂糖水	4,760 kg	14,280 kg
ビタミンC	480 kg	1,440 kg

**《操作》**

操作名	調合釜1	調合釜2
混合1	15分	15分
冷却	15	15
混合2	10分	10分

**【製品規格】**

項目	規格値	測定方法
$\times \times$	$\times \times \times$	JIS $\times \times$
		JIS

PFC

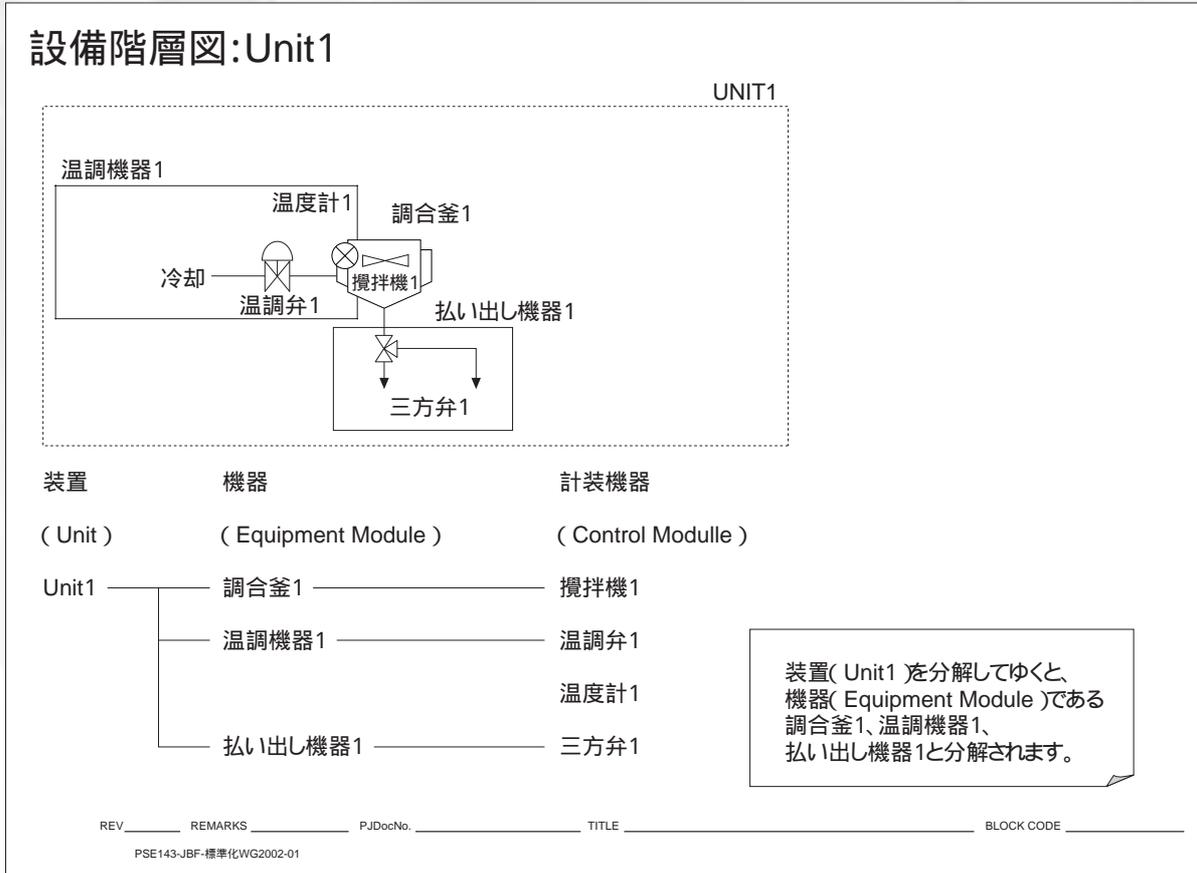
REV \_\_\_\_\_ REMARKS \_\_\_\_\_ PJDocNo. \_\_\_\_\_
TITLE \_\_\_\_\_
BLOCK CODE \_\_\_\_\_

PSE143-JBF-標準化WG2002-01

例 3.3 リンゴジュース製造処方仕様

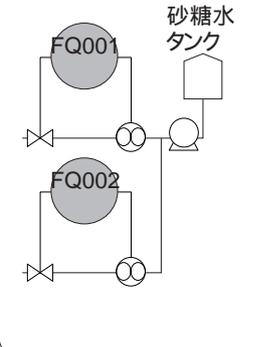
## 設備ベース仕様

例3.4は、プロセスモデルで定義された製品を実際に製造するためのハードウェアの面から階層化したものです。ソフトウェアの汎用化、再利用性のためには機器( Equipment Module )のくくり方が鍵となります。



例 3.4 設備階層図

例3.5は、ひとつの機器制御の仕様を記述したものです。書式として左側に入力、右側に出力を示し、中央部に必要な処理を記述しています。他の機器制御についても同様に記述していく必要があります。

INPUT	PROCESS	OUTPUT
<b>HMI 入力</b> 自動・半自動モード切替SW From 他unit/Equipment module 処方パラメータ Unitからの仕込み開始指令 仕込量 中断再開指令 フィールド入力信号 砂糖水タンク液切れ ポンプトリップ ポンプ起動信号 流量計1,2信号 弁開アンサ信号	<b>機能</b> EM1 砂糖水仕込み機器制御  <p>シーケンス</p> <ol style="list-style-type: none"> <li>1.ライン確保</li> <li>2.バルブ開</li> <li>3.ポンプ起動・アンサ確認</li> <li>4.積算開始</li> <li>5.積算カウントアップ</li> <li>6.バルブ閉</li> <li>7.ポンプ停止</li> </ol> <p>複数ユニット( unit1 , 3 )からの同時要求に対し並列実行可能。ポンプ停止は、全ての要求がないことを確認し停止する。</p>	<b>HMI 出力</b> 運転モード表示 運転状態表示 計量実績レポート TO 他Unit/Equipment module Unitへ仕込み終了通知 仕込み異常終了通知 計量実績データ通知 <b>収集実績データ</b> 仕込み先 仕込み実績値 <b>フィールド出力信号</b> ポンプ起動・停止指令 弁開閉指令 <b>警報処理</b> 仕込み量異常 ポンプトリップ 砂糖水タンク液切れ
REV _____ REMARKS _____ PJDocNo. _____ TITLE _____ BLOCK CODE _____ PSE143-JBF-標準化WG2002-01	<b>異常処理</b> ポンプ停止 バルブ閉 バッチカウンターリセットは行わない (オペレータによる手動リセット)	

例 3.5 機器制御仕様

## 共通仕様

S88 で部分的に触れられている制御システムを構築する上で必須の共通項目を示します。

例 3.6 は、シーケンス状態遷移を表したものであり、工程の“状態”に対し“押しボタン”などによりどのような状態に遷移可能かを表したものです。

## オペレーション仕様

### モード

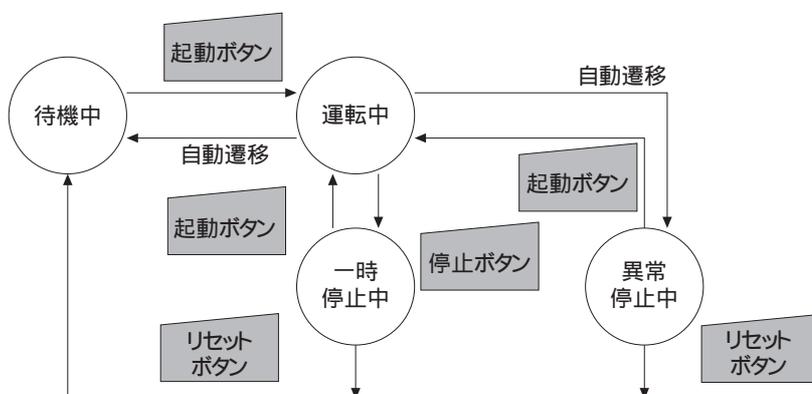
各ユニット・Equipment Moduleは  
モードとして自動・半自動・手動の3つのモードを持つ。

### 状態遷移

各ユニット・Equipment Moduleは  
状態として下記4つの状態を持つ。

### 手動介入

常時(どのモードであっても)手動介入操作可能とする



REV. \_\_\_\_\_ REMARKS \_\_\_\_\_ PJDocNo. \_\_\_\_\_ TITLE \_\_\_\_\_ BLOCK CODE \_\_\_\_\_

PSE143-JBF-標準化WG2002-01

### 例 3.6 オペレーション仕様

例3.7は、自動シーケンスの運転中に異常停止やオペレータにより緊急停止操作時の制御装置での対応の考え方を記述しています。

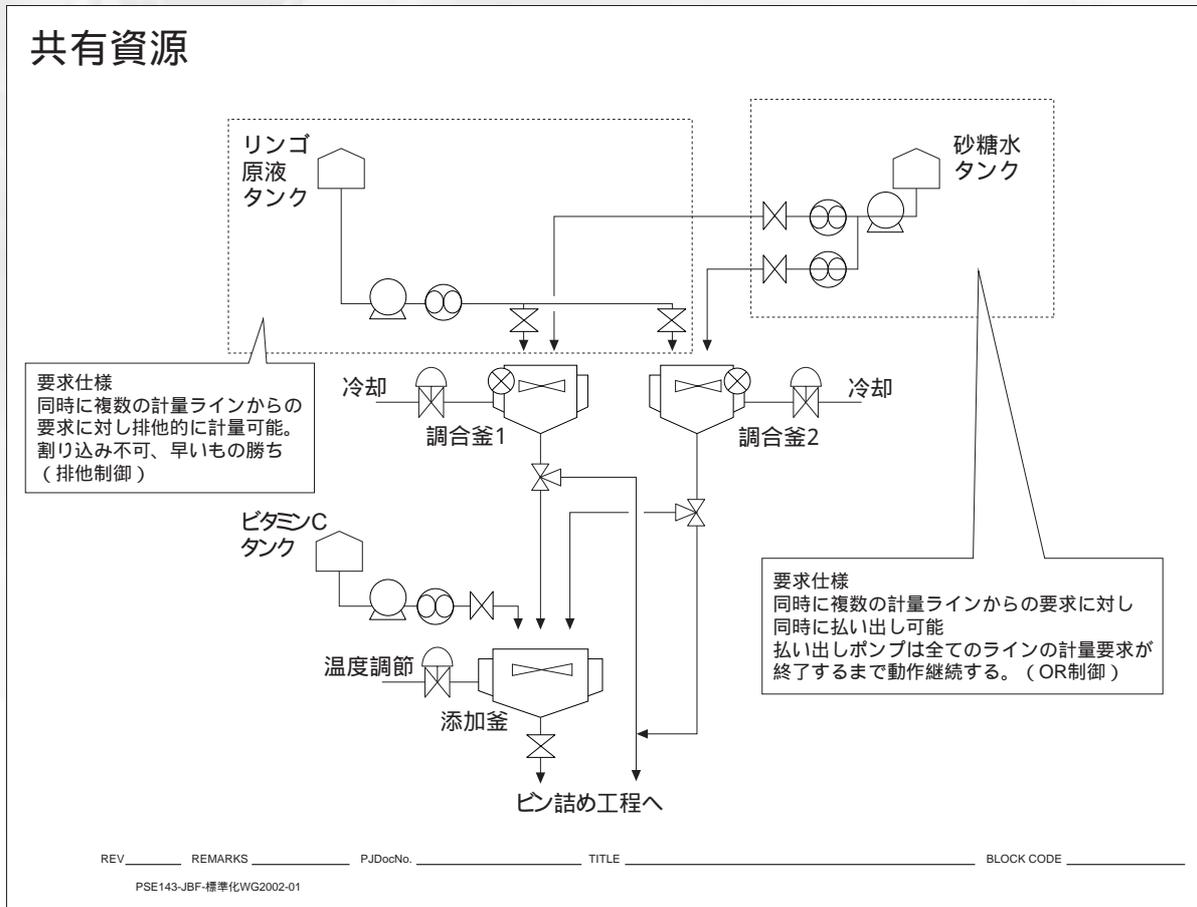
## 異常処理の考え方

1. 異常とは、自動シーケンス渋滞ならびにオペレータからの緊急停止ボタン操作により、自動シーケンスが緊急処理を行わなければならなくなった状態を指す。  
手動介入による例外操作は含まない。
2. 異常状態緊急処置  
制御システムからの出力オフ(回転機停止、バルブは閉等プロセスの安全サイド)
3. 異常状態からの回復手段  
オペレータ責任により、異常状態復旧後に再開指令を受け該当工程の頭より再開する。
4. 停止状態の処理  
異常停止とは異なり、制御システムからの出力はせず、そのままの状態を保持する。

REV \_\_\_\_\_ REMARKS \_\_\_\_\_ PJDocNo. \_\_\_\_\_ TITLE \_\_\_\_\_ BLOCK CODE \_\_\_\_\_  
PSE143-JBF-標準化WG2002-01

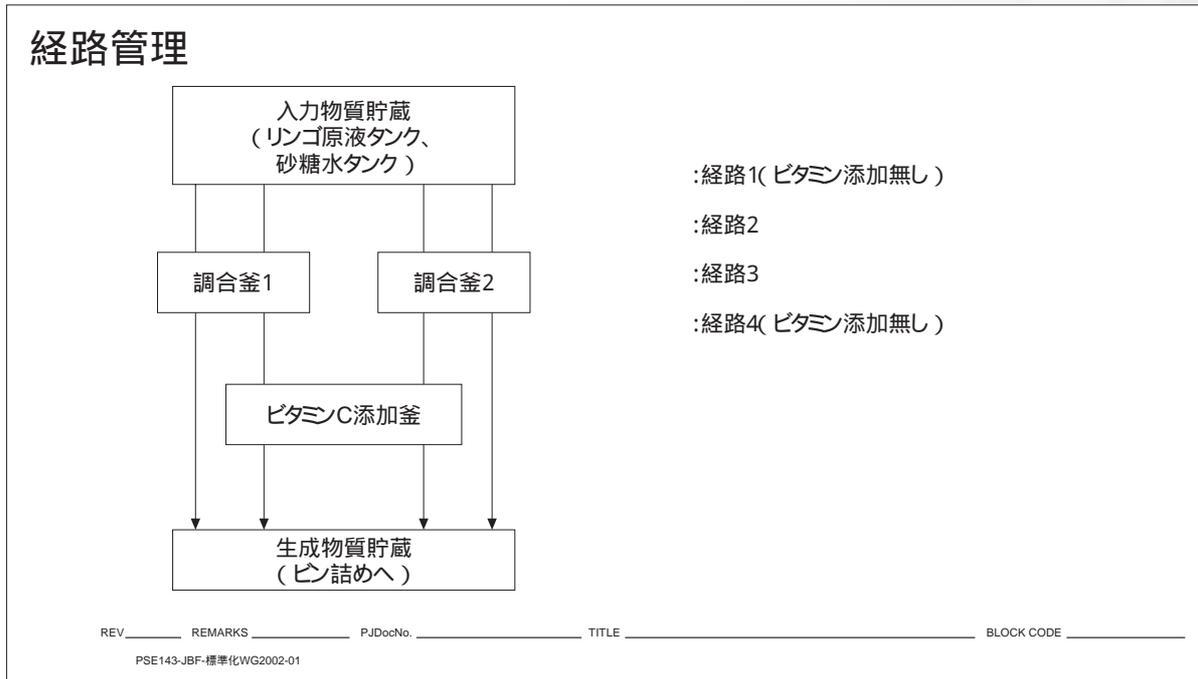
### 例 3.7 異常処理仕様

例3.8は、複数の装置から使用される機器を共有資源として整理したものです。  
 この例では、リンゴ原液は排他的に使用し、砂糖水は同時使用を許しています。  
 例3.8の他に共有資源となる機器1つ1つに対し、待ち行列制御等の仕様を記述する必要があります。



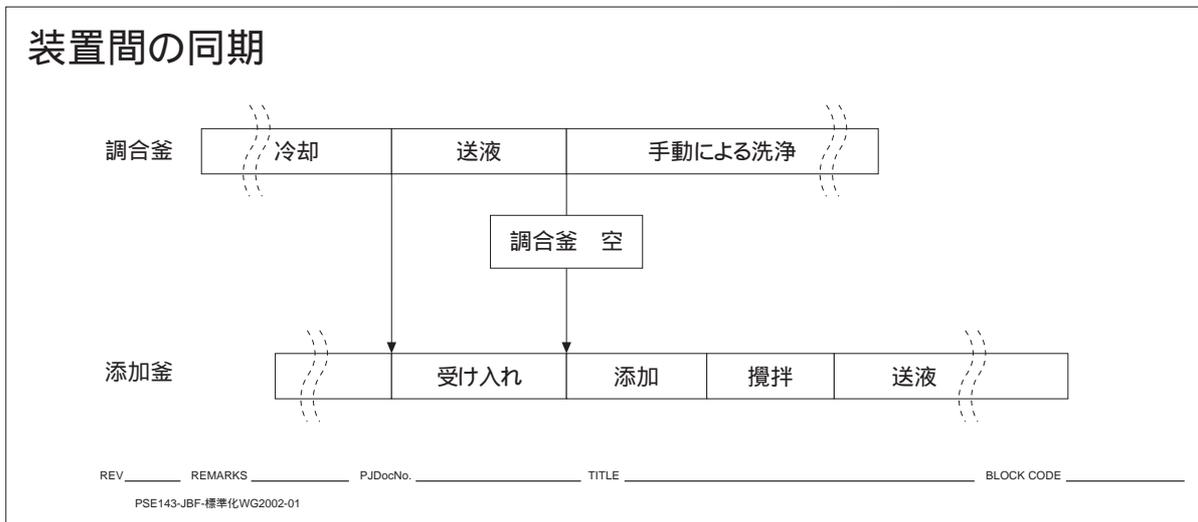
例3.8 共有資源管理仕様

例3.9は、ひとつのバッチが物理的制約条件やその他の諸条件を考慮した上でどのような装置を通過して流れていくかといった経路を示したものです。



例 3.9 経路管理仕様

例3.10は、各装置の中の単位操作間で物質の受け渡しのタイミングをガントチャートで表したものです。



例 3.10 装置間の同期仕様

## おわりに

「S88とはなにか?」、「どのようなメリットがあるのか?」について多少なりともお分かり頂けたでしょうか。S88の狙いの一つは、2章で述べたとおり、再利用性にあります。そのために、モデル化手法を用いて、バッチプラントを“プロセス”、“物理的な設備”、“手順制御”の異なる次元(視点)に分けて階層化し、それらのマッピングを行っています。これによって、“製品に依存している製造の手順”と“設備に依存している制御”を明示的に分けることを可能にし、再利用性の問題を解決しています。ところが、S88を勉強する側にとってみると、このモデリングの部分こそが、S88を理解するうえでの最大の障壁であるといっても過言ではありません。そこで、本書はこのモデリングの部分を分かりやすく説明することに努めています。

一方、S88はモデリングの為の“How-to”を示したものではありません。標準として、モデル化のための視点を与えた、といった方が正しいでしょう。例えば、図2.3に、「装置(Unit)の分割」と題して物理モデルの階層化の例を示し、「バッチ制御でのソフトウェアの汎用化、再利用性を考慮すると、機器(Equipment Module)のくくり方が鍵となります。」と説明しています。しかし、「どのように機器をくくったら良いか」については、S88には書かれていませんし、それを目的としてはいません。ある意味で、個別の実装に対応してはユーザーに任せ、再利用性の実現に向けたモデリング対象の捕らえ方、基本的な考え方を規定することを目的としています。

本書が、モデリングを中心に説明していると述べましたが、入門書としての性格上、モデリングに関しても多くの部分を省略せざるを得ませんでした。また、その他の部分(例えば“生産情報”、“例外処理”、“管理アクティビティ”、“処方管理”、“生産管理”、“スケジューリング”)については、仕様書例に記載されているものもありますが、殆ど触れていません。更に「分かりやすく」するために、大胆な解釈を加えた部分もあります。その意味で、本書を読んでS88を理解するのではなく、S88を読破し理解するための第一ステップとして頂きたいのです。そして、そのとき本入門書がお役に立てれば、幸いです。

なお、付録Aには、参考として日本規格協会発行 JIS C 1807 バッチ制御-第1部:モデル及び用語(2002)解説部分からの抜粋を記載しました。また、付録Bには、より高度な応用として『商品開発プロセスモデルの考察』(寄稿)を加えました。

最後に、S88入門書の発行までの長期に渡るワーキンググループ活動をささえていただきましたJBF(Japan Batch Forum)の世話人及びメンバーの皆様、JBFの上部団体でありますプロセスシステム工学第143委員会の委員長をはじめとする幹事の皆様ならびに日本学術振興会のご理解とご支援に感謝いたします。



# 付録

付録A	バッチ制御 - 第1部:モデル及び用語 解説 (日本規格協会発行 JIS C 1807 バッチ制御- 第1部:モデル及び用語(2002)からの抜粋)	32
-----	--	----

---

付録B	寄稿 『製品開発とレシピエンジニアリング』 名古屋工業大学 伊藤 利昭	36
-----	--	----

---

### 1. 制定の経緯及び趣旨

バッチ制御システムに関する国際標準は、ISA( International Society for Measurement and Control )が中心にして標準化作業が行われてきました。1988.10 ISAにSP88( Standard Practices88 )が創立され、それに対応するように IEC TC65/SC65A にWG11 バッチ制御システムが作られました。

1995年にISA S88.01 Batch control Part1: Models and terminologyが規格化され、これを受けた形で、1997年にIEC 61512-1 Batch control Part1: Models and terminologyが制定されました。そのような中、日本学術振興会 プロセスシステム工学第143委員会 ワークショップNo.2( 運転管理のためのバッチプロセスのモデリング )/WG3 標準化分科会は、IEC 61512-1の有効性を評価し、モデル及び用語を標準化することのもたらすメリットからJIS化への要請を、(社)日本電気計測器工業会に行ないました。これを受けて、IEC TC65/SC65A/WG11の国内対策委員会の窓口である、(社)日本電気計測器工業会は、バッチシステム規格化検討WGを設け、日本語化の作業を進めJISとして制定されました。IEC 61512-1の序論には、この規格の目的、意図として次のことが述べられています。

この規格で定義されるモデル及び用語は、次のことに使われます。

- バッチ製造プラントの設計及び運転に関して行うべきことを明確にすること。
- バッチ製造プラントの制御の改善に使うこと。
- 自動化の度合いに関係なく適用すること。

特に、この規格がバッチ製造プラント及びバッチ制御に関する標準用語、一貫した概念とモデルとを提供することで、関係する当事者間の意志疎通を改善し、次のことを可能にします。

- 新製品が完全な生産レベルに到達するまでの時間を短縮すること。
- ベンダーがバッチ制御を実用化する適切なツールを提供すること。
- 使用者が要求を十分に規定すること。
- 制御システム技術者の補助無しでも処方を開発できるようにすること。
- バッチプロセスを自動化するコストを削減すること。
- ライフサイクル全般にわたるエンジニアリング負担を削減すること。

また、この規格は次のことを意図していません。

- バッチ制御を導入又は適用する唯一の方法が存在すると示唆すること。
- 使用者にバッチプロセスで現在使用中の方法を捨てることを強制すること。
- バッチ制御分野での発展を阻害すること。

この規格の中で提示されたモデルは、完結したものであると仮定していますが、モデルは縮退又は拡張しても構いません。しかし、装置と計装機器との階層は、物理モデルから省略することはできません。同様に基本処方と実行処方を、処方のモデルから省略することはできません。これらモデルを縮退、拡張するための特定の規則は、この規格には含まれません。

- 縮退：モデルを構成するある要素は、モデルが整合性を保ちかつその要素の機能が残される限り、モデルから削除してもかまわない。
- 拡張：ある要素をモジュールに付加してもかまわない。もしモジュールが関連した要素間に付加されるときは、元の関係の整合性が維持されなければならない。



管理業務モデルと他のモデルと関係についてこの規格では規定していません。図A.2は管理業務モデルと物理モデルとの位置関係の一例です。

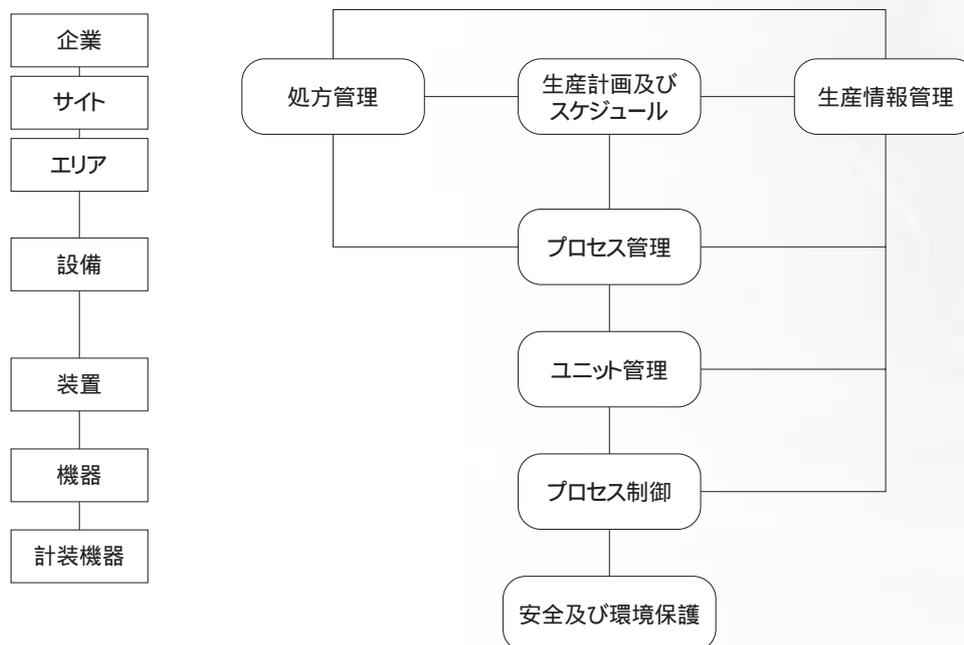


図 A.2 管理業務モデルと物理モデルとの位置関係例

### 3. 4種類の処方とその対応分野

バッチ製造プラントにおいて、製造手順とプロセスとを関連付けるのが処方であり、処方には4種類があります。上位の原処方及びサイト処方は、研究・企業段階及びサイト段階に対応しており、具体的な物理設備を想定していません。これに対して下位に位置する基本処方及び実行処方は、具体的な物理設備を想定しています。

- |                          |                     |
|--------------------------|---------------------|
| 1)原処方( General Recipe )  | ..... 具体的な設備を想定しない  |
| 2)サイト処方( Site Recipe )   | ..... 具体的な設備を想定しない  |
| 3)基本処方( Master Recipe )  | ..... 具体的な設備を想定している |
| 4)実行処方( Control Recipe ) | ..... 具体的な設備を想定している |

原処方とサイト処方との処方手順がプロセスモデルに従っており、基本処方及び実行処方の処方手順は手順制御モデルに従っています。

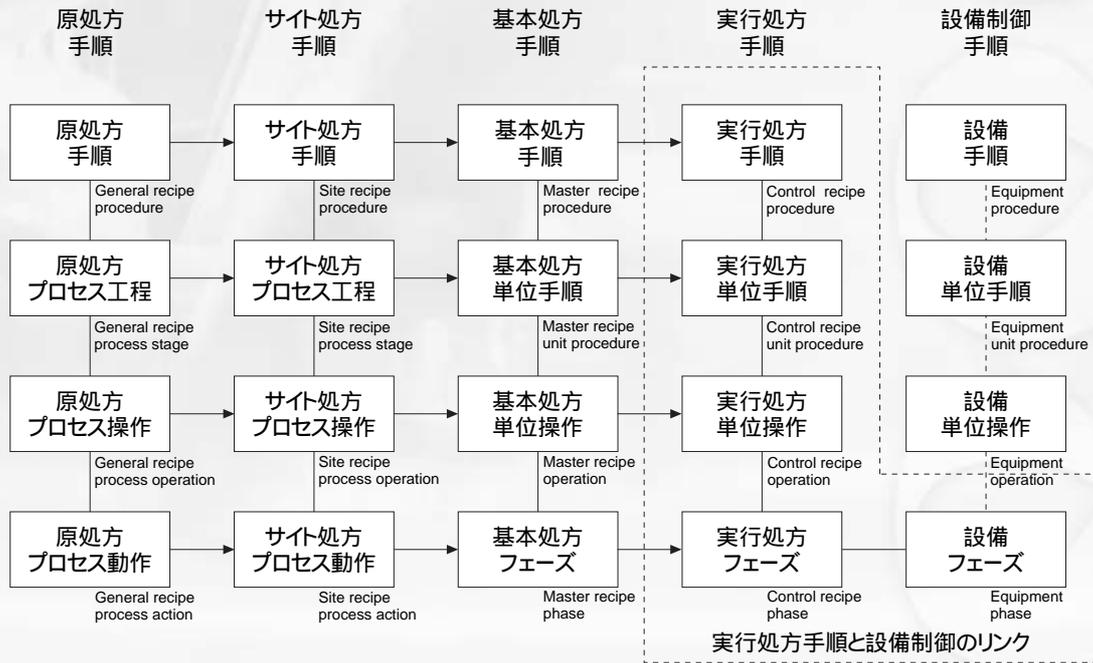


図 A.3 処方手順と設備制御との関係例

### 1. 製品寿命の短命化とエンジニアリング業務の変化

表B.1に示すように、この10年間に製品の寿命は大幅に短くなりました。たとえば、化成品の場合、10年前には製品寿命が14.1年もありましたが現在では5.7年になっています。このように製品寿命が短くなると設備投資の回収が難しくなるので、新製品生産のために新プラントを建設するのではなく、既存のプラントを利用するケースが増えています。

これに伴いエンジニアリング業務の内容も大きく変わりつつあります。プラントを新設する場合には、建設を中心としたエンジニアリングが重要ですが、既存プラントを利用して新製品を生産する場合には、レシピを中心としたエンジニアリング(レシピエンジニアリングと呼ぶ)が重要になります。すなわち、既存プラントをベースとして新製品のレシピを作成し、それに基づき必要なプラントと制御システムの改造をおこないます。

この場合には、レシピエンジニアリングを含む製品開発期間の短縮をはかることが戦略的に重要です。製品開発期間を短縮できれば、他社に先駆けて新製品を市場に投入することができ、新規市場で有利なポジションを獲得できます。また、早期に新製品を市場に投入することにより新製品の寿命を実質的に延長できるので、定量的な効果として売上増を期待することができます。この効果は、製品寿命に較べて製品開発期間の長い場合に顕著で、たとえば医薬品の場合には製品寿命が平均9年ですから、製品開発期間13.2年を10%短縮することは、製品寿命を15%伸ばして10.3年とすることに相当し、1.3年分の売上増を期待できます。

表 B.1 製品寿命の短命化と製品開発期間の短縮状況<sup>1</sup>

	製品開発期間(年)			製品寿命(年)		
	10年前	5年前	現在	10年前	5年前	現在
食料品	2.2	2.0	1.7	6.4	4.4	2.8
化成品	5.0	3.9	3.0	14.1	8.5	5.7
医薬品	9.9	10.8	13.2	15.8	12.0	9.0
洗剤・化粧品・油脂	7.0	4.7	2.8	7.7	6.7	5.7
ゴム製品	2.0	1.3	0.6	5.5	4.0	2.5
家電機器	1.6	1.5	1.1	1.6	1.3	0.9
半導体・デバイス	4.2	3.2	1.9	5.4	3.8	2.9
情報・通信機器	3.8	2.5	1.6	4.8	3.4	2.0
自動車	4.7	3.4	2.3	7.3	5.3	4.6
全事業分野平均	4.3	3.6	3.0	11.1	8.9	8.1

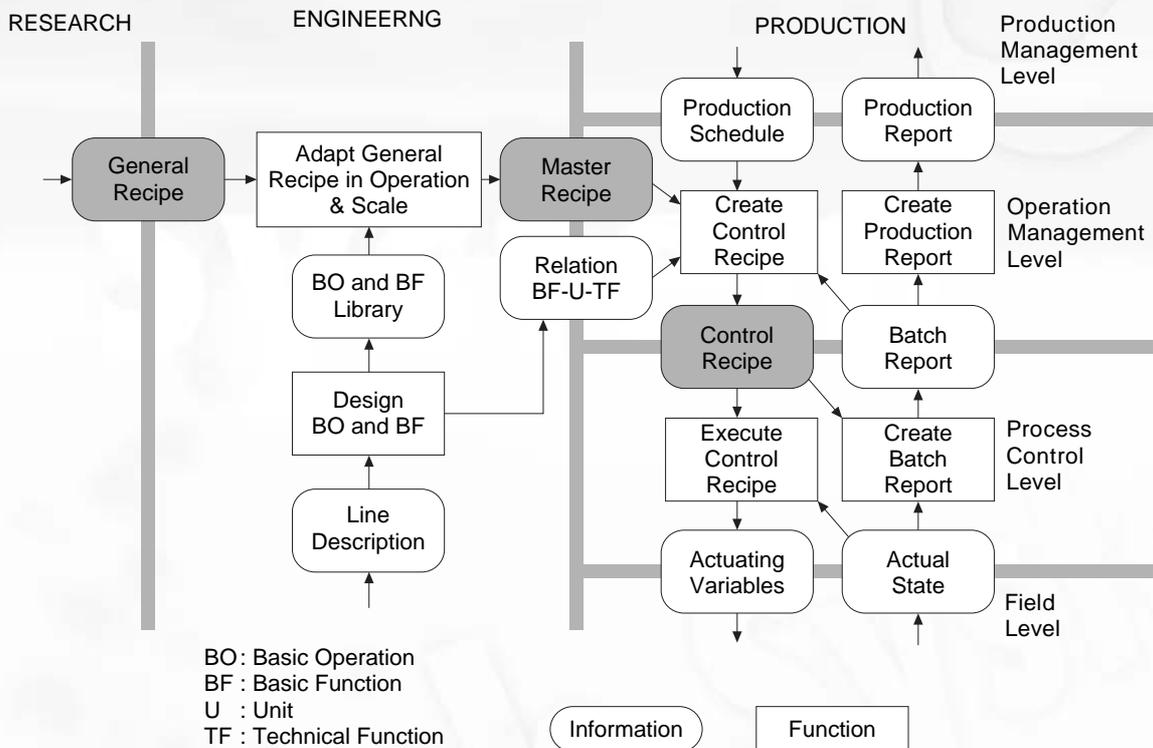
1 経団連「産業技術力強化のための実態調査」報告書(1998)から抜粋

## 2. レシピエンジニアリングとは

ライン河畔に立地する化学会社の計装・制御技術者の集まりであるNAMUR<sup>2</sup>は、1992年にRequirements to be met by systems for recipe-based operations( レシピを基本とした運転のための制御システム仕様 )を発表しました。NAMURは、このなかでレシピが決まっていく過程と制御システムの間を関係を図B.1のように整理しています<sup>3</sup>。このなかには三つの重要な概念が含まれています。

第一は、製品を開発する過程を、General Recipe、Master Recipe、Control Recipeという3種類のレシピで表現したことです<sup>4</sup>。General Recipeは、研究開発段階で作成されるレシピです。General Recipeにはプラントに関する情報は含まれていません。これに対して、Master Recipeはエンジニアリングの段階で作成され、特定のプラントでの製造を前提にした製造処方です。また、バッチプロセス用制御システムで実行する製造処方であるControl RecipeはMaster Recipeを詳細化したもので、実際のプラントの状況( 機器がメンテナンス中など )も反映されています。

NAMURはこのように研究開発 エンジニアリング 製造に至る製品商品開発過程を3種類のレシピを用いてモデル化しました。



図B.1 NAMURのRecipe-Based Operationモデル

- 2 NAMUR was founded at Leverkusen on November 3, 1949, as the body to represent the interests of the users of measurement and control technology in the chemical industry by such renowned experts in the field as Dr. Sturm (Bayer), Dr. Hengstenberg (BASF) and Dr. Winkler (Hüls). (<http://www.namur.de/index.html>)
- 3 NAMUR-Empfehlung: Anforderungen an Systeme zur Rezeptfahrweise, NAMUR AK 2.3 Funktionen der Betriebs- und Produktionsleitebene (1992)
- 4 S88では、General RecipeとMaster Recipeの間にSite Recipeを定義している。

第二は、Master Recipe や Control Recipe が自動合成できる可能性を示したことです。

図 B.1 では、Master Recipe は General Recipe に書かれている製造手順と製造設備の対応する機能を適合させてつくることになっています。具体的には、General Recipe に書かれている製造手順(図 B.1 中の BO: Basic Operation)を製造設備の対応する機能(図 B.1 中の BF: Basic Function)を照合して製造設備の運転手順に置き換え、General Recipe に書かれている Formula を製造設備の諸元に合わせてスケールアップまたはスケールダウンすれば Master Recipe をつくることができます。

また Control Recipe は次のようにしてつくることができます。まず、生産スケジュールで指定される生産量から、原料・助剤などの必要量を決定し Formula<sup>5</sup> を完成させます。つづいて製造設備の各機器の状況(定期点検中、故障中など)を考慮して使用機器を決定します。そして、Master Recipe に書かれている製造設備の運転手順を各機器の操作レベル(図 B.1 中の TF: Technical Function)まで詳細化すれば Control Recipe が出来上がります。

図 B.1 に示されているようなレシピの生成が、自動的にあるいはコンピュータの支援のもとで容易に行えるようになると、研究開発段階で得られる General Recipe から短時間で Control Recipe が生成できるようになり、製品開発期間の大幅な短縮を期待できます。またレシピの変更管理もシステム化されるので、社内の業務の合理化だけでなく、品質や安全上の事故やトラブルを防止でき、顧客から大きな信頼を得ることができるでしょう。

第三は、バッチプロセスにおける生産管理の考え方を整理したことです。図 B.1 の Production の欄では、4 つのフィードバックループを構成することにより生産管理がおこなわれています。

Field レベルでは、通常の温度制御などのフィードバックループやバルブの開閉指令とリミットスイッチによる開閉確認信号を対にしたようなフィードバックループが構成されています(フィードバックループ # 1)。

Process Control レベルでは、バルブの開閉状況などをみながら、レシピの実行が管理されます(フィードバックループ # 2)。この機能と Field レベルの機能を合わせたものが、一般的なバッチプロセス制御システムの機能です。

Operation Management レベルでは、バッチレポートがレシピと照合されます。すなわちレシピどおりの生産が行われたか否かが確認されます(フィードバックループ # 3)。確認の結果、たとえば昇温時間がレシピより長くかかった場合には伝熱面の汚れが想定され、洗浄を検討します。また製品品質に偏りがみられる場合には、Formula の調整が行われます。

このように、レシピは生産管理の中核です。なお、Production Management レベルでは、図には書かれていませんが、生産報告が生産スケジュールと照合されます(フィードバックループ # 4)。予定通りの生産が行われなかった場合には、生産スケジュールを変更することになります。

以上をまとめると、レシピエンジニアリングでは、レシピを核にして研究開発 エンジニアリング 製造に至る製品開発過程のシステム化をはかることと、レシピを中核とした生産管理と制御のシステム化をはかることが主要なテーマになります。

5 JIS C 1807 バッチ制御 第 1 部：モデル及び用語(2002)では、Formula を処方パラメータと訳している。原材料、エネルギー、人的資源など製品の生産に必要な資源の名前と量(プロセス入力)、温度、圧力、時間などの製造条件(プロセス変数)、処方の実行により生成される物質、エネルギー、副産物などの名前と量(プロセス出力)を指す。

### 3. レシピエンジニアリングに関する技術課題と研究・開発状況

レシピエンジニアリングに必要な技術やツールは、まだ確立しているわけではなく開発途上にあります。主な技術課題と研究開発状況は下記のとおりです。

#### レシピ手順の合成

レシピ生成過程における手順の合成は、製造経路と使用装置の決定<sup>6</sup> 移送手順の合成<sup>7</sup> 各装置での処理手順の合成<sup>8</sup> 例外処理手順の合成<sup>9</sup>に分類でき、国内外で多くの研究がおこなわれています。また、これらを支援するシステムも開発されつつあります<sup>10</sup>。

#### Formulaの変換則

同じ品質の製品を装置規模や仕様の異なる装置で製造するためには、運転条件の変換則を求める必要があります。変換則が確立できれば、実験をしなくてもFormulaのスケールアップとスケールダウンができるようになります。

#### レシピに適應する制御則

Control Recipeごとに変化する生産量や製造条件によって制御ループのパラメータを変更できれば、よりよい制御を実現できるはずですが、モデル予測制御はプラントの動的なモデルを内蔵しているので、生産量や製造条件を制御ループのパラメータに反映させることができます<sup>11</sup>。

#### バッチプロセスのスケジューリング

レシピは一つのバッチを対象としたものです。実際には同一プロセス内で多くのバッチが独立して進行しますから、バッチ間の協調をとらねばなりません。たとえば、装置から装置へ移送するためには装置間の同期が必要です。計量タンクや排気・排液処理装置の様な共有資源は、バッチからの要求に応じて割り当てが制御されねばなりません。あるいは、装置の洗浄のように各バッチには属さないが製品品質に大きな影響を与える操作もスケジュールに織り込まねばなりません。このようにバッチプロセスのスケジューリングは設備の効率や環境・安全・品質の確保に大きな影響を与えるので、国内外で多くの研究がおこなわれています<sup>12</sup>。

6 星, 山下, 鈴木: バッチプロセスのグラフ表現に基づく操作手順生成アルゴリズム, SICEシステム情報部門学術講演会論文集, (2001)

7 濱口, 橋本, 伊藤, 米谷, 戸苅: バッチプロセスのための自律分散制御系, SICE 制御部門大会資料 495-498 (2002)

8 橋爪, 矢島, 小野木: 階層ベトリネットを利用したバッチ制御のための操作手順の合成, SICEシステム情報部門学術講演会論文(2001)

9 日本学術振興会 PSE143 委員会 WS20 報告書「運転・管理のためのバッチプロセスモデリング」(1999)

Bajagopalan Srinivasan and Venkat Venkatasubramanian: Automating HAZOP analysis of batch chemical plants, Computers Chem. Engng, Vol.22, No.9, 1345-1355 (1998)

10 青山, ホッサムガパール, 仲: 顧客志向生産のための情報モデル, コントロールレシピ自動生成, コントロールレイヤー, 顧客志向生産システム研究会研究報告書 3-17(2003)

11 江口, 野口, 小崎, 伊藤: バッチプロセスの最近の技術動向と将来展望, M&E 4月号(2003)

12 Shinji Hasebe: Production Planning and Scheduling What is the Real World? Proc. of the Int. Symp. on Design, Operation and Control of Next Generation Chemical Plants (PSEAsia2000), pp.617-622 (2000)

#### 4. 産業の枠を越えたレシピエンジニアリング

図B.2に示すように、新製品を開発するためにはいろいろな企業や産業の協力が必要です。 のサプライチェーンが企業の枠を超えたマネジメントシステムとして発展しつつあるのと同様に、 のレシピエンジニアリングも企業や産業の枠を超えて発展させることが大切です。特に、日本の産業技術の強みである「摺り合わせ」の強み<sup>13</sup>をシステム化する際には、レシピエンジニアリングが重要な役割を果たすことになるでしょう。

レシピエンジニアリングはバッチプロセスに限定されたものではなく、ディスクリートプロセスや連続プロセスにも適用されるべきものです。産業や企業の枠を超えて多くの方々がレシピエンジニアリングに関心をもっていただけることを期待します。

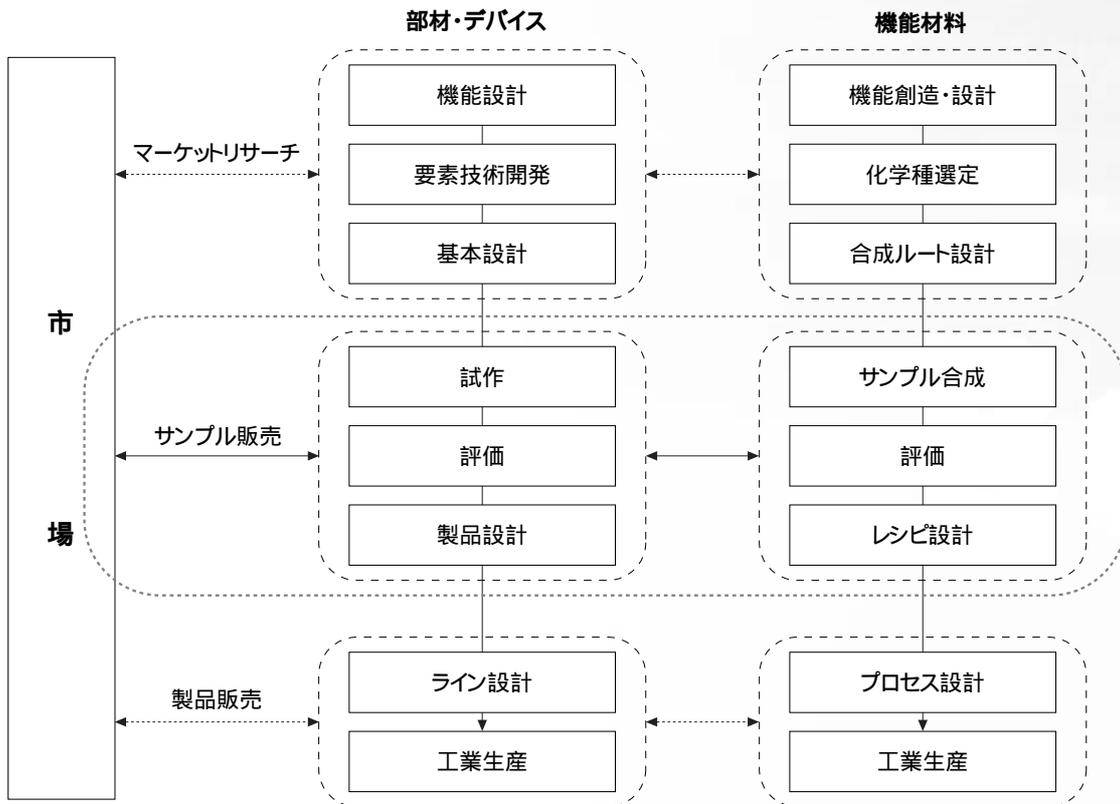


図 B.2 Cross-industry Recipe Engineering

13 機能性化学産業研究会：新たな企業・産業文化の形成による価値提案型産業への挑戦 (2002)

## 参考文献

日本規格協会発行 JIS C 1807 バッチ制御 第1部：モデル及び用語(2002)

日本学術振興会PSE143委員会WS17報告書「バッチプロセスの設計・運転・管理のための体系的アプローチ」

日本学術振興会PSE143委員会WS20報告書「運転・管理のためのバッチプロセスモデリング」(1999)

Jim Parshall, Larry Lamb : Applying S88 Batch Control from a User's Perspective:ISA(1999)

ANSI/ISA 88.00.03-2003

Batch Control Part 3: General and Site Recipe Models and Representation Approved 14 March 2003

本書の御利用にあたっては、以下に御同意下さい。

- ・ 本書の著作権は、日本学術振興会プロセスシステム工学第143委員会常設分科会ジャパンバッチフォーラムに帰属します。
- ・ 再配布可能です。
- ・ 内容を改変しないでの引用は出典を明示( Japan Batch Forum: S88入門、2004年3月 )の上で許可します。
- ・ 営利目的でのご利用は御遠慮下さい。



独立行政法人 日本学術振興会  
プロセスシステム工学 第143委員会  
常設分科会

**ジャパン バッチ フォーラム(JBF)**

<http://www-jbf.pse.ce.tuat.ac.jp/>