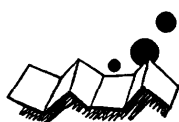


解説

衛星システムにおける信頼性技術†



下 平 勝 幸††

1. はじめに

大規模システムの典型的な代表例が、人工衛星とその周辺を含むシステムである。打上げ用ロケットとその関連システムまでを含むと非常に大きなシステムとなる。人工衛星の開発は、常にこのロケットシステムの能力の制約を受け、ときには機能、能力などの調整に苦しむこともある。制約の点では地上システムと異なった性質をもち、より厳しい環境条件、修理不能／高い信頼性、重量の制約、増加するコストの低減という中において人工衛星を設計、開発しなければならない。

人工衛星は目的機能をもって打上げられるが、そのためのミッション機器とそれを支援するバス機器とで構成される。運用期間が長く、修理不能のシステムであり、両機器のバランスをとりながら、無欠陥を指向しなければならない。そのためほとんどのバス機器はバックアップ、すなわち冗長系をもたなければならない。ときには一つの部品の信頼性向上で対応する必要もあろう。そのような設計条件を合理的に調整するために信頼度及び故障モード解析などが用いられる。

多くの打上げ経験から、故障の多くが潜在的な欠陥によるものであると考えられ、事前に情報があつたとしたら防ぎ得たものといわれている。とすれば、それら経験情報をもれなく収集し、再発防止を目的として原因を解析し、確実に次のプロジェクトに反映することが、結果として目標達成につながる。このような故障解析技術は、今後の宇宙プロジェクトに必要な要件といわれている。

また近年の人工衛星の機能の増加、高機能化は目をみはる。そのため LSI を用いたマイクロコンピュータ化は必須であり、システム運用上このソフトウェアの信頼性確保はきわめて重要である。

以上の状況にあつて宇宙開発、特に人工衛星の信頼性保証についての状況、技術、体系を紹介することは、一般の大規模システム構築にとって有意義と考え、以下のようにまとめた。

2. 人工衛星の開発

人工衛星は、ミッション機器とバス機器で構成される。人工衛星の目的別の代表例として、通信衛星、放送衛星、地球観測衛星があげられる。バス機器は、あくまで従であつて、ミッション機器の機能と信頼性を維持、支援するものであるが、ほとんど一体として開発しなければ効率的でない。図-1 は、バス機器の代表的構成を示している。

構造系は衛星の骨格を成し、機器を保持して人工衛星を形作るほか、太陽電池やアンテナ、センサなど衛星の本体外にも機器を設置する役目をはたす。姿勢安定の方式によって形は変わり、二つの方式に合うよう設計される。図-2 にスピンによる安定方式と、能動的な方式によって常に必要方向へ衛星を安定させる三軸安定方式の例を示した。

人工衛星のエネルギーは、太陽電池によって供給を受ける設計が一番に多く、その電力の電圧、配分を制御

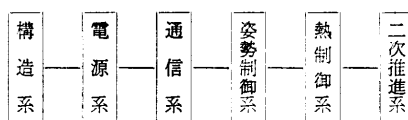
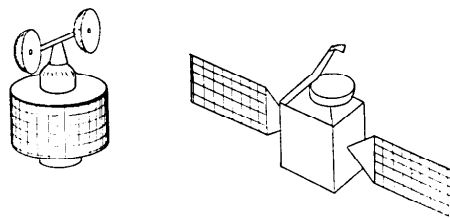


図-1 典型的な人工衛星の構成



スピン安定

三軸安定

図-2 代表的な人工衛星の形

† Reliability Engineering for Satellite System by Masayuki SHIMODAIRA (National Space Development Agency of Japan).

†† 宇宙開発事業団

したり、二次電池に蓄える機能を電源系という。大きな電力を必要とした場合には、三軸方式が有利である。地上からの指令を受け、それを解読し、必要箇所へそれを配分し、また衛星内の所定データを収集、配列して地上へ送る機能を通信系といい、またテレメトリコマンド系ともいっている。解読、配列、蓄積などの機能が徐々に複雑となってきたために、ここにはコンピュータが使用されはじめています。衛星の姿勢を安定化させたり、軌道の修正を行ったり、所定方向へ向けるための機能を姿勢（軌道）制御系という。ここにも精度、機能の増加からコンピュータが使用される。真空中で太陽光の有無による温度差が激しい人工衛星は、熱を制御する機能を有し、機器の温度幅をできる限り少なくすることが信頼性を確保する上できわめて重要であり、これを熱制御系という。熱は受動的に制御するか、または能動的に制御するかが、長期にわたって少ない温度幅範囲とするためには、大型衛星はほとんど両者、特に能動的な制御を採用している。姿勢や軌道を制御する場合にはアクチュエータの力を借りなければならないが、中でも燃料を使ったエンジン型式のアクチュエータを二次推進系と呼び、ガスジェットなどの技術が用いられる。

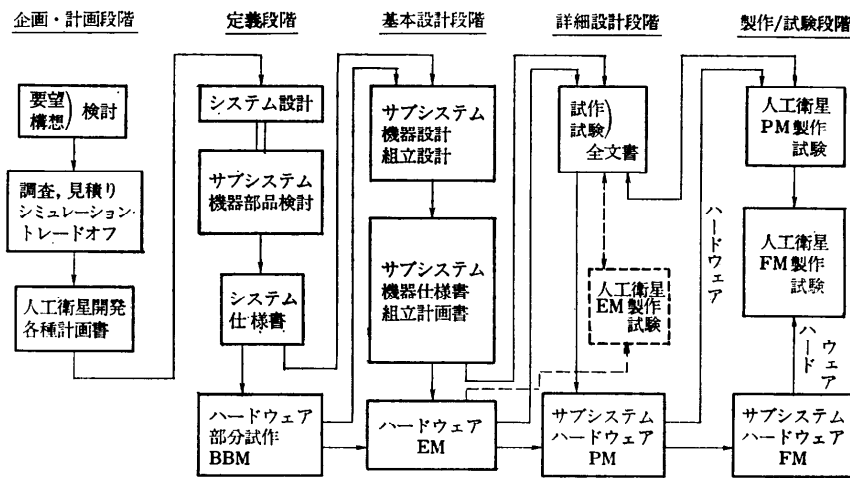
人工衛星は、ロケットの打上げ能力、人工衛星を収納するフェアリング容積によって制限され、また性能は、必要条件としての軌道、安定性、取得できる電力、希望する寿命によって調整され設定される。ミッション機器側の寿命、重量、電力、安定性の要望によってバス機器は設計されるが、前述の制限のほか、State of the Arts によって、開発期間、経費とのバランスで妥協しなければならない。すなわち人工衛星のすべてのパラメータは、十分初期に調整し、無欠陥を目標として設定されなければならない。このような初期の検討作業を企画、計画段階または概念設計段階と呼んでいる。以後、着実にベースラインを設定し、それを実現させるための設計作業を行って、アウトプットを出すとともに必要なら元のベースラインを変更させることになる。企画・計画段階で見積もられた計画は、そのシステムを構成する要素を検討することによってより具体的なシステムの定義、すなわちシステム要求条件を設定できる。その段階を定義段階または予備設計段階と呼んでいる。このように段階を追ってベースラインを設定し、必要な修正を合理的に行うことをコンフィギュレーション管理という。

予備設計段階でシステム仕様書（すなわち人工衛星

開発仕様書）が設定されると、それを実現させるための設計に入る。このフェーズでは、人工衛星のすべての構成が決定され、基本的な設計は完了するとともに細部の設計、仕様書が設定される。このフェーズを基本設計段階という。予備設計段階では実証されていない未開発ハードウェアまたはサブシステムは、すでに設計支援用試作モデル（ブレッドボードモデル、Bread Board Model—BBM）が作られ、機能確認が行われて設計が決まっていく。基本設計段階でも、機器などで製造文書が十分整備されていないハードウェアに対しては、技術モデル（Engineering Model—EM）が作られ、機能、耐環境性、製造/加工/組立文書、検査文書が作成される。

以上のような情報がまとまるとあとは人工衛星としての製作文書の整備が残る。これを整備するのが詳細設計段階である。ハードウェアについても、システム組立の前のフェーズとして、EM 文書を作製する公式試作試験がある。これをプロトタイプモデルまたは認定モデル（Prototype Model (PM)/Qualification Model (QM)) という。このモデルはそのまま人工衛星の試作時に用いられ、衛星としての PM の一部として使用される。このフェーズを製作/試験段階という。製作/試験を通じて多くの問題や欠陥が指摘され、修正されるが、その PM ハードウェアが所要の機能をもっていることを確認するとともに、すべての文書が修正され、正規なものとして確認される。この文書に従って製作するのがフライト用の実機である。このモデルは原則的にはストレスを受けず、品質確認すなわち、PM 設計文書どおりに作られ、製作上の欠陥のないことが確認されるだけで十分フライト価値/能力があると判断される。またはそのように管理しなければならない。図-3 は以上の人工衛星の開発の流れを示している。（フェーズの呼称については、理解しやすくするため宇宙開発事業団で用いている用語と異なる表現とした。）

人工衛星の設計フローについてはほとんどこの流れに変更はないが、ハードウェアについてはかならずしもこの図どおりのフローで設計、製作、試験されるとは限っていない。たとえば、実績がなく、全く新規の設計の機器、または人工衛星であれば、このフロー及び必要な他のモデル、たとえば、配列を検討するモックアップ、熱の設計に必要なデータをとる熱モデル、振動などの構造、製作設計のための構造モデル、またはそれらの部分モデルなども追加され、EM 及び



BBM: Bread Board Model EM: Engineering Model PM: Prototype Model FM: Flight Model

図-3 人工衛星開発フロー

PM を通じてフライト用の人工衛星が製作される。その逆に実績のあるハードウェア、または設計技術を採用し、経験的にも十分信頼性が保証される場合には、BBM, EM を通らず PM のみか、またはその PM の試験の後、必要なチェックと修正を行い、フライトに供する方法を採用しうる。また実験的な機器では時に EM をそのままフライトすることもあり、事情に合わせてプランをたてる。しかしこの場合あくまでもフライトのハードウェアは、宇宙環境及び所要寿命をもち、他の機器に対して悪い影響を与えないよう保証されなければならない。

3. 信頼性管理と技術

人工衛星に対する信頼性要求は、次の三つに集約される。

- (a) 信頼度
- (b) 寿命
- (c) 信頼性設計の考慮

(a)の信頼度は、ミッション寿命要求に対して、どの程度の残存確率となるかを示すパラメータで、一般的には、経験則的に決められるが、最近5. で述べる SYROP が開発されたことから、性能とのトレードオフで決められるようになり、最も経済的でリスクの少ない開発を指向した信頼度の要求値が設定できるようになってきた。図-4 に信頼性技術の適用例を示す。

信頼度を用いる目的は、目標とする信頼性を達成するために、どんな構成とするか、その冗長系採用の

レベル及び部品の選定、品質保証、適用などを決定するためのもので、かならずしもフライトの信頼度を確実に実証させるためのものではない。そのために信頼度のデータのほとんどは MIL-HDBK-217()^{*}を用い、そのほか、みずからの試験データとフライト実績データ、地上の実績データなども用いる。図-5 に目標値設定から設計への影響を与える流れを示す。このうち冗長系の基本形は 図-6 に示すとおり一般の技術と全く同じである。LSIを用いる系では、LSIにエラーが発生することは当然ありうるとして、かならず多数決冗長方式を採用する。また冗長系についてはかならず FMEA (Failure Mode and Effect Analysis)/FTA (Fault Tree Analysis)^{**}によって本当に冗長系となっているかを分析する。またエレメント（または部品）の故障によってシステムが機能停止するようなエレメントを単一故障点 (Single Failure Point—SFP) と呼び、次の二つのうちいずれかの処置をとる。一つはそのエレメントで冗長系を採用して SFP を除去する。またもう一つの方法は、十分に設計マージンをとったり、品質保証を完璧にとることによって高信頼化を図る。いずれにするかは改善の効果をもって判断する。このような品目を開発中の信頼性管理品目と指定し、製造から打上げまでの間、品質を保証する特別手

^{*} エレクトロニクス、エレクトロ・メカニカル部品の故障率を環境条件、適用条件、品質保証条件などを指定することによって求められるようハンドブック化した米国政府の文書。

^{**} FMEA: 構成する部品の予想される故障のモードを仮定し、それがシステムにどのように影響するかを解析する手法。
FTA: システムの故障モードを取り上げその要因をリストアップし、その影響度合を解析する手法。

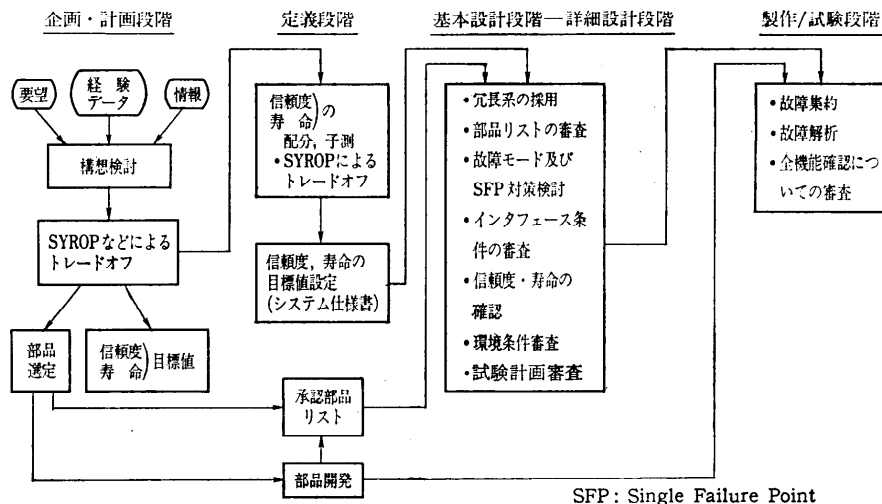


図-4 信頼性技術の適用例

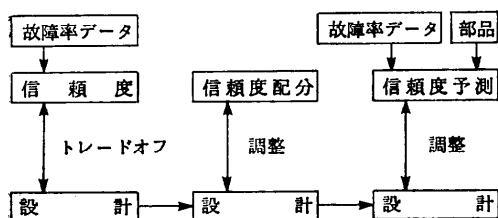


図-5 信頼度と設計の関係

- ① 冗長系の採用/構成 ② 部品選定/回路設計
③ 部品の品質保証 ④ 部品の適用/環境条件

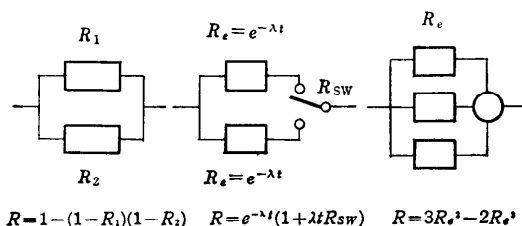


図-6 衛星でよく採用する冗長系の基本形

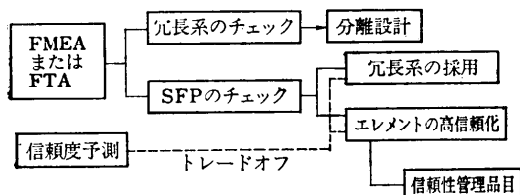


図-7 FMEA/FTA と必要な処置

当をとる。図-7 に、FMEA/FTA とそのとるべき処置について示した。

信頼度は、部品によっても大きく変わる。部品の取付部の温度、負荷条件、品質保証、デューティサイクルなども影響し、これらをすべて管理することによってユニットの信頼度が決定される(図-8)。すなわち信頼度というパラメータを用いて、部品の品種、調達条件及び適用をすべて管理/設計することになる。

以上は信頼度を中心とした解析、設計技術であるが、もう一つ重要な技術として FMEA または FTA がある。FMEA の目的は次の三つに限る。

- (a) インタフェースの確認
(b) 冗長系の確認
(c) 共通部に対する故障の影響

設計には予測しない過誤が残る。それを設計文書のすべてをチェックして除去しようとしても無理であり、どうしても試作まで至ってしまう。なかでもインタフェース部はエラーが集中することから、正常時のインタフェースの確認のみでなく、予想される異常条件を

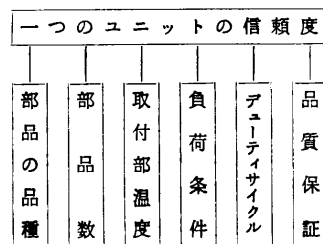


図-8 ユニットの信頼度の決定要素

A ユニット → B ユニット

部品	異常	インタフェース点	端子番号	影響
トランジスタ	短絡 発振 直流 重量		P101 23ピン	制御停止 " 対策済

図-9 インタフェース FMEA の例

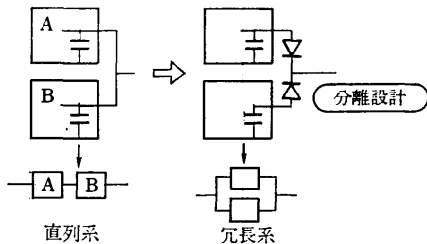


図-10 FMEA による冗長系の確認例

仮定し、それによってもユニット、機器、サブシステム間のインタフェースが適当であるかを確認する。これをインタフェース FMEA と呼び、その例を図-9に示す。

冗長系は確かに信頼性向上の最適手法ではあるが、実際の設計上ではきわめて難しい技術である。図-10に一つの例を示すが、油断すると直列系となるため、冗長系については、FMEA によって確かな冗長系となっていることを確かめなければならない。またバス電源、構造など共通部に継がる各ユニットは、そのユニットが故障しても、システムが機能停止とならないよう、共通部に対する各ユニット故障を FMEA によって解析し必要な対策をとらなければならない。

寿命は使用する部品や素材によって決定される。一般に低軌道の周回衛星は、2～3年、静止衛星で5～10年が要求される。これは単に要望によって決まるのではなく現実のハードウェアが制限となる。一番クリティカルな寿命制限部品は二次電池である。続いて燃料、半導体の放射線による劣化、衛星の外に付けられた熱制御材料の劣化などが並び、ミッション機器のカソードヒータ、リーク電流、ランプ、ベアリングなどが続く。これらの部品や材料の劣化特性は、信頼度計算の場合の故障率データと異なり、実際に使用する部品で決定しなければならない。そのために加速によって時間を短縮して評価するものと、実時間の寿命試験によって評価するものに分けられる。実時間によるものとして、ベアリング、コンデンサ、ヒータなどがあ

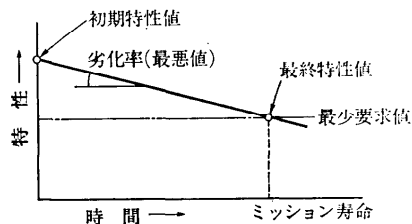


図-11 特性劣化の例

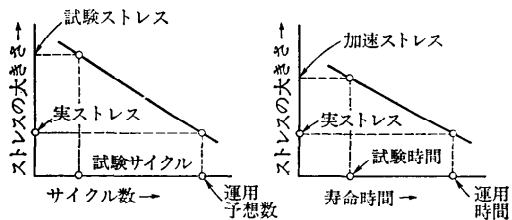


図-12 時間、サイクルによる寿命推定

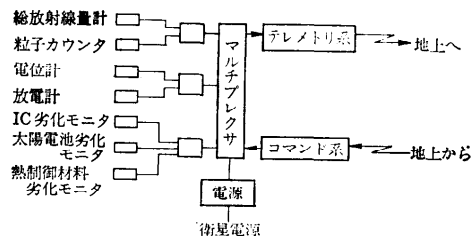


図-13 TEDA の構成

げられ、再度の寿命保証が困難なため、できるだけ一度評価したら、その基本技術は変更しないようにして部品やユニットを設計、適用する。図-11に初期特性値の設定の例を図-12に実運用での寿命を推定する方法を示したが、これらの基礎となるデータは、多くの試験の結果から求められる。そのため実運用でどのようなストレスを受けるか、その環境条件を十分知らなければならない。わざわざそれら工学データをとるために観測装置を人工衛星に搭載する。その装置をTEDA (技術データ取得装置) とよび、図-13のような機能をもっている。

人工衛星は、部品や材料をもって組み上げられ、故障の単位が部品または加工点であることから、信頼性技術の一つとして部品、加工技術の管理が含まれる。故障の原因を分類すると

- (a) 破壊 (オーバストレスによる)
- (b) 拡散 (金属などの移動)

- (c) 腐食 (酸化, 電食などを含む)
- (d) 疲労 (繰返し応力)
- (e) 結晶 (結晶化による破壊原因)
- (f) 汚染 (外的要因による)
- (g) 発熱 (摩擦, 熱集中)

などがあげられることから, 部品自体に原因をもつこともあれば, 使用法が不適, 予想以上の条件, 仕様未決定などをその原因として追及するかまたは予想による事前対策をとっておかなければならない. 一般に十分な設計と試験によって潜在的欠陥をとり, 後天的な欠陥を試験によって除去した部品では, 適用法を間違えないかぎり故障はない. また人工衛星の信頼性保証は, 熱設計に依存し, 十分な温度マージンを部品や加工部に対してとってあれば, これも効果大きい.

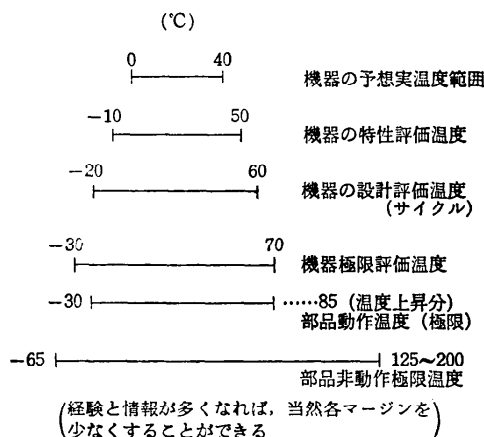


図-14 機器と部品の温度範囲関係例

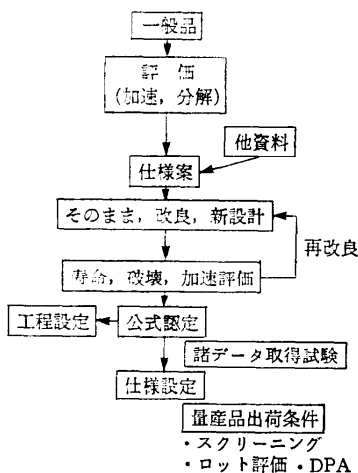


図-15 部品開発フロー

このように, 宇宙用部品は, 仕様を決め, 部品内の設計, 製造法を管理し, 試験によって欠陥品を除去し, ロット保証を加速試験, DPA (Destructive Physical Analysis) 分解物理解析 (または良品解析) によって行えばきわめて高い信頼性が保証される. 図-14 は部品の温度設定の流れを, 図-15 は部品開発の流れを示してある.

人工衛星の開発に当たっては, 種々の技術を適用するが, 最終的には無欠陥を指向する. 図-16 に開発の流れの中での信頼性に関する主要業務をあげた. 設計審査の本来の目的は, ベースラインの確定であるが, 同時に代替の提案を審議し, 問題点を討議することも目的に含まれる. 当然技術常識, 社内基準, 過去の人工衛星での経験についても検討され, 本当にこの設計でよいのかを審議する. したがって基準や過去の経験情報が多いことによって, その出力の信ぴょう性は高まる. 同時に試験, 運用中の問題点はすべて報告され, 検討される体系が必須である. 図-17 は, 不具合発見から最終の有効情報としてフィードバックするまでのフローを示した.

図-17 で処置とあるのは, 対象となるシステムのその場での処置を指し, 故障解析を待って最終決定されることを意味する. 故障解析は, FTA, 現物解析, シミュレーション実験, 再現試験までを含む. すべての不

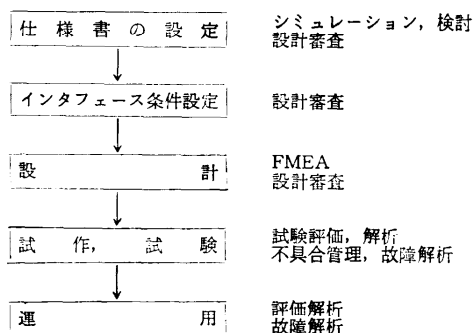


図-16 開発の流れでの主要業務

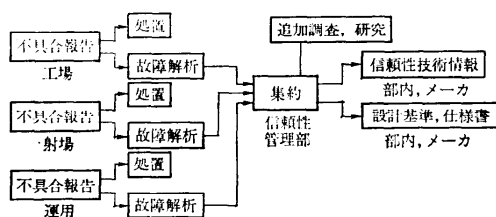


図-17 不具合とその活用

具合について解析の対象とするのではなく、機能上の欠陥を通常対象とする。その情報は信頼性管理部に集められ、追加調査、試験を受けて、当面緊急に宇宙開発事業団の全部門、メーカにその経験を配布する。とともに恒久的情報については、設計基準または仕様書に反映し、同一事故を再発しないよう、その情報を有効活用する。この情報は、一点一点の情報が多く、かならずしも統計量として表現されるものではない。それは、宇宙開発が無欠陥を指向するからである。

4. オンボードコンピュータ

人工衛星は2. で述べたようなサブシステムで構成され、以前の小型の人工衛星では考えられなかった高度な機能が各サブシステムに要求されるようになってきている。きわめて精度の高い姿勢制御、有効な電力配分、多くのデータの蓄積と地上への送信、地上からの多くの命令の処理など、とうてい従来のアナログ、またはロジック回路では処理できない情報を取り扱わなければならない。そこで最近の人工衛星ではマイクロコンピュータが採用されてきている。現在の人工衛星で採用されている例は次のサブシステムである。

- (a) 姿勢または軌道の制御
- (b) コマンドのデコーディングと配分
- (c) テレメータデータの配列または蓄積と伝送
- (d) 星の観測による方位計算
- (e) 太陽電池パドルの回転制御

今後は、人工衛星、それに類したプラットフォーム、宇宙移動体、宇宙船などにはますます採用されるであろうし、採用しなければならなくなるものと予想される。

現在のハードウェアは、データ処理系が8ビットを採用し、姿勢制御系が16ビットを採用しているが、LSIはすべて宇宙の環境に耐えることが要求され、特に、放射線に対する耐候性、電源電圧変化に対応できる安定性及び幅の広い温度条件に耐えられる耐力などがあげられる。したがって民生品のままではほとんど採用できず、相当な改造が行われている。

放射線に対しては、地球磁場に捕捉された電子及び陽子によって半導体内の絶縁膜の劣化が生じ、徐々に機能を失うトータルドーズ（総線量）効果、及び高エネルギー陽子及び宇宙線、特に重イオン粒子がLSIを通過したときに生ずるフリップフロットまたはゲートのステータスが反転する、いいかえれば宇宙線によ

て情報が変化する Single Event Upset 現象の両方を考慮しなければならない。また放射線も軌道によって大きく変わるため、一率に高いレベルの耐力を有するLSIを採用する必要もない。従来の衛星開発では、トータルドーズレベルの高いLSIが宇宙用として必要であるとしてきたが、このアップセットが近年報告され、宇宙用としてはこのアップセットのない素子でもなければならないことになってきた。その発生量はLSIの設計によって左右され、最近はその品種も少しずつ出てきてはいるが、やはり多少は認めざるをえない。すなわち素子ではその対策にも限界があることから、コンピュータまたはコンポーネントとしてビット反転の存在を認めた設計、構成にしておかなければならない。すなわち使用する冗長設計は、2 out of 3 またはそれと同等の構成が要求され、いろいろ工夫されている。図-18 ではその代表例として、三つまたは4つのコンピュータユニットで構成し、常に二つのユニットの出力を比較して一致したデータを使用する方式を示した。

使用するソフトウェアは高級言語を用い、一般に実証された、支援ツールの整備されたものを選定する。近年人工衛星を含む宇宙機システム間でデータを交換したり、宇宙においてシステムがドッキングする計画が検討されはじめた結果、CPUを含むデータフォーマット、使用する言語までも統一または標準化することが提案されてはいるが、今のところ言語としても、またハードウェアとしても決定したものはなく、もう少ししばらくそのために時間を要するものと思われる。現在は二つの流れがあり、CPUとして2901を代表とする4ビットスライス型を使用したフレキシビリティをもったマイクロプログラム方式と、8085または8086シリーズを主流とした実証型の採用によって開発期間の短いプロジェクトの危険を回避しようとする流れである。現在のところはデータの伝送上の約束以外には統一はとらず、当面は一般技術の進歩と宇宙活動の要件を備えた方式が採用されていく。

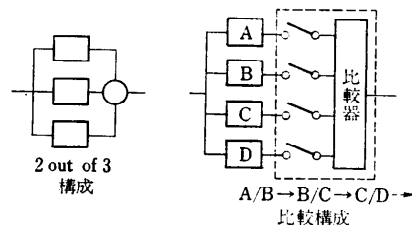


図-18 コンピュータの構成例

ソフトウェアは、リアルタイム処理であるためエラーを打上げ前に十分デバッグし、システムとして欠陥のないものにしておくことが必要である。また人工衛星の開発は計画から打上げまでの期間が5年、7年と要し、打上げてから低軌道で2～5年、高い軌道で10年の運用を要求されることから、常にデータ処理に対する要求が変化し、それにソフトウェアが対応できるようにフレキシビリティと容量をもっていなければならない。

ソフトウェアの開発は、まず要求される外部条件の設定が十分できていなければならない。その条件は、人工衛星に対する要求条件であり、処理ダイナミクスの整理である。概してこの外部要求が十分に調整されていなければ、以後の開発計画に無理が出たり、コストの上昇と欠陥を含む結果になってしまう。現在開発し、昭和66年次ごろに打上げを計画している人工衛星のコンピュータ及び搭載ソフトウェアの開発はすでに58年次から開発に着手し、数年にわたって外部要件の検討と、シミュレーションによる人工衛星運用の試験を続けており、その間にも外部要件が常に変わっていくことを経験している。このことは、要求条件の完全なる管理が必要であることを示唆しており、定義ならびに必要ならば変更していく、コンフィギュレーション管理の必要性を示している。すなわち外部要件は、アルゴリズム、フローチャート、ダイナミクス、シーケンス、時間、ユニット間のインタフェースなどを文書化し、その約束のみが守られ、書かれていない条件はすべて対象としない仕様書方式によってベースラインを設定するのがよい。以後設計、試験を通じて常に変更管理し、人工衛星が打上げられるまでソフトウェアの産物とこの仕様書が常に一致していることが必要である。

図-19 にソフトウェア開発の時間配分の例をあげたが、前述のとおり相当長期にわたる開発となり、かならずしも設計、開発、検証の区別はしにくい。そして中でも検証に要する時間と費用は、当初から見積もられていなければならない。オンボードソフトウェアの開

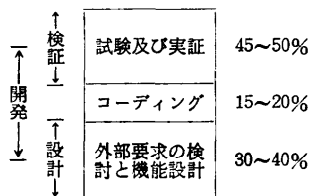
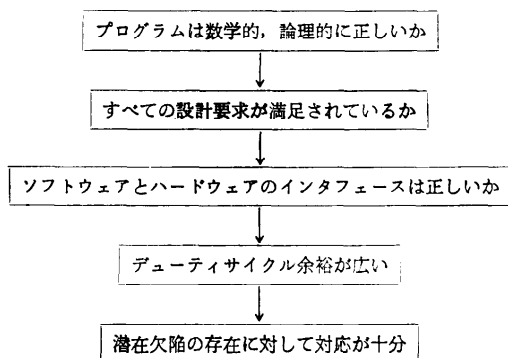


図-19 ソフトウェア開発の時間配分



ソフトウェアの信頼性を得るためコンピュータ入力条件を大きくし、厳しいストレスを与える。
(入力データ: 大きな外乱、センサ、アクチュエータの変動など)

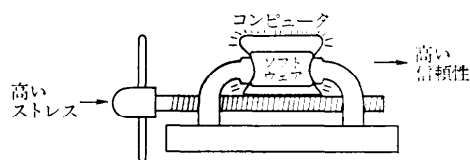


図-20 検証の概念

開発はここに相当費用を投入しなければならないからである。そして効果的な開発方法とするため、評価方式は、予定のダイナミックレンジよりも幅広くとることが有効である (図-20)。表-1 にオンボードソフトウェアの検証目的をあげた。結果として運用に至るまでの間に多くの検証の機会と必要ツールを準備し、その工程での能力に合った確認を行うのが順当であろう。設計及び支援ツールを図-21 にリストアップした。もちろん汎用ツール及び人工衛星設計に共通なツール以外は、そのプロジェクトまたはサブシステム開発に合わせたプログラムであり、品質保証のための必要なツールである。そして人工衛星は種々の種類や特定の機能を要求するが、基本となるアルゴリズムは多くなく、ソフトウェアは基本はモジュール構造にしておいて、特定の人工衛星の要求に合うよう、条件、インタフェースを調整し、全体を設計する方式がとられる。これによって設計を簡素化するとともに、確実な開発を行うことに役立てる。そしてフライト実証は実機で行う以外に方法がないため、フライト前に十分な品質保証が必要となる。

図-22 に開発のフローと品質保証の内容を示した。初期の機能設計では、姿勢制御理論の設計と制御定数

表-1 オンボードソフトウェアの検証

工 程	検 証 目 的	検 証 項 目	検 証 対 象
システム設計	・アルゴリズム	・機能, 性能	・アルゴリズム
ソフトウェア設計	・外部要求 ・機能の設計	・数学モデル ・インタフェース	・数学シミュレーション プログラム
単体試験	・ソフトウェアモジュール	・能力機能	・モジュール
総合試験	・ソフトウェア	・モジュール間インタフェース ・開ループ, 閉ループ機能 ・外部条件 (入出力, 異常) ・時間, 容量	・オンボードソフトウェア
コンピュータを含む 電気性能試験	・インテグレーション	・タイミング ・開ループ機能 ・入出力	・ソフトウェア ・オンボードコンピュータ
サブシステム試験	・オンボードソフトウェア ・インタフェース	・外部インタフェース ・閉ループ機能 ・初期条件	・ソフトウェア ・オンボードコンピュータ ・入出力インタフェース ・センサ, アクチュエータ
システム試験	・人工衛星としての機能	・メモリダンプ ・開ループ機能 ・入力条件	
運用 (定常)	・定常機能	・テレメトリ監視	・運用サブシステム
運用 (異常)	・ソフトウェアモジュール ・アルゴリズム ・パラメータ	・オンボードとシミュレーション の相互比較 ・メモリダンプ	

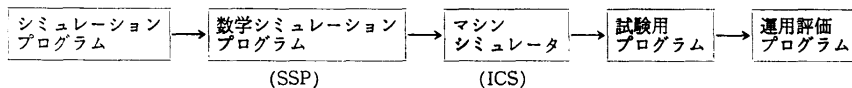


図-21 搭載プログラム開発ツール

設定のために、数学シミュレーションプログラム (SSP—Scientific Simulation Program) が用いられる。これによってアルゴリズム及び条件が確認できるが、これは以後、オンボードソフトウェア及びコンピュータシステムのテスト時における基準系として用いられる。試験Ⅰのフェーズにおいては、ソフトウェアのみの検証を目的とし、コンピュータハードウェアを模擬した ICS (Interpretive Computer Simulation Program) を汎用中型コンピュータに搭載し、それにオンボードコンピュータソフトウェアを搭載してセンサデータ、コマンドデータを入力し、アクチュエータデータを出力させて SSP データと比較検証をコンパレータによって行う。またハードウェアが存在すれば、同一データを用いてベンチテスト装置を使用し、同様に計算して出力を比較検証を行うこともできる。これはオープンループによる精度照合とタイミング照合をもってソフトウェアの機能を確認することを目的とする。この段階の試験条件は、規定値内及び相当広い範囲にわたるコンティンジェンシ条件を時間の許すかぎり多数設定し、試験を行うことが必須であり、正

常条件だけの機能確認のみを行うことだけでは不十分である。

試験Ⅱにおいては、ソフトウェアを搭載した ICS と、同様にソフトウェアを搭載した実コンピュータシステムを構成し、SSP と比較しながら、エミレータによる入力と、実センサを用いた入力条件とによりクローズドループ試験を行う。エミレータによる場合を静クローズドループ試験、実センサによる場合を動クローズドループ試験という。このフェーズは一般に小規模で行い、実センサをフライトテーブルや温度チャンバなどに搭載しフライトシミュレーションを主とした試験を行い、欠陥を除去する。

試験Ⅲにおいては、システム全体を構成し、大規模化させ、同様に実コンピュータと試験用プログラムによるクローズドループ試験を行う。実コンピュータは静クローズドループ及びフライトシミュレーションを模擬したテストスタンド上のセンサを用いた入力を用い、正常及び異常、及びダイナミックレンジを広げた条件を処理することをもって検証を受ける。これによってすべての系は確証を受け、人工衛星へ搭載され

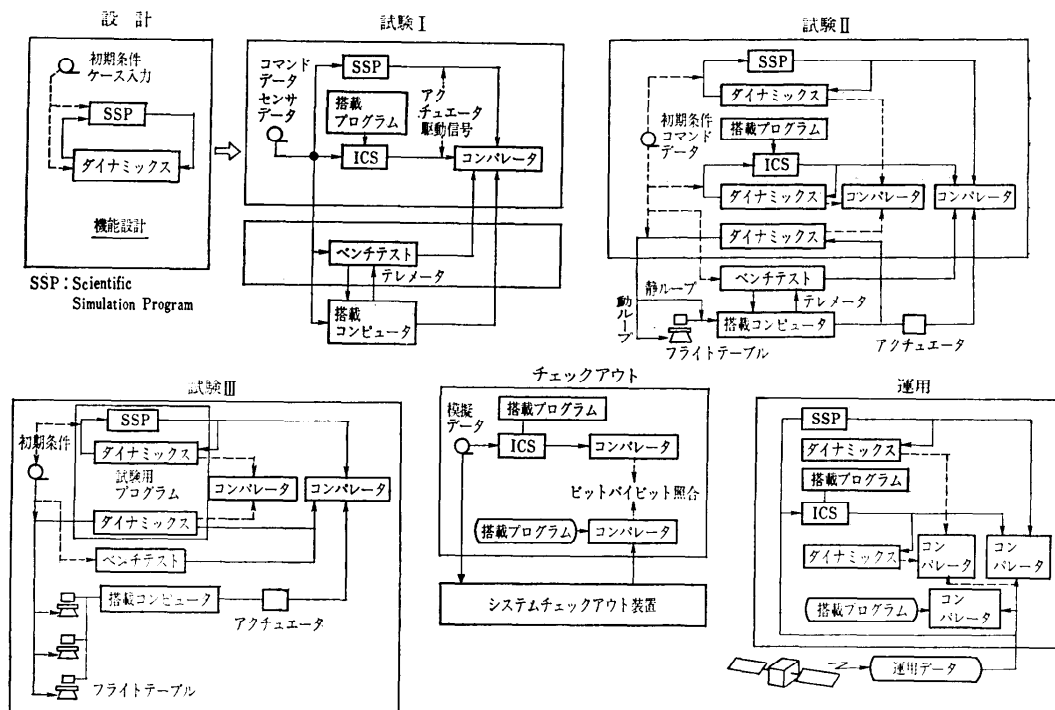


図-22 搭載プログラム品質保証フロー例

る。データ処理系は通常搭載された状態で実際に運用され、その後も検証を受ける。姿勢系においては、搭載された状態では簡単な機能チェックしか受けられない。そのため、搭載後も ICS を用い、地上チェックデータを入力し、ビットバイビットの異常有無の確認などを行う。

人工衛星は地上からの制御を受けて運用される。衛星側にも、搭載されたソフトウェアによる自動制御が機能する。これらを総合して運用を模擬した試験が、運用手順書、地上の運用コンピュータシステムの検証ならびに要員の訓練が行われる。このフェーズにおいても、ICS や実コンピュータ、エミレータ及びフライトテーブルなどの実シミュレーションが用いられる。

このようにして一般にコンピュータが搭載される場合にはすべてのフェーズで検証やシミュレーションが実行され、単なるソフトウェアを対象とするのではなく、常に人間や実の人工衛星条件を含めたシステムとしての機能が確認されている。

5. 信頼性設計プログラム

2. で述べたように、人工衛星はロケットの打上げ

能力によって制限され、その中でどこまで機能を組み入れるか、設計で苦慮するところである。このことは、そのまま信頼性設計で制限を受けることをも含んでいる。一つの例として、同一の打上げ重量の中で、できるだけ多くのミッション重量を搭載しようとしたとき、無理をしてもバス機器の重量を減少させるため、高度の機能を有する部品を採用し、多少バックアップ機能の少ない設計（すなわち小規模の冗長設計）を採用したり、熱設計の重量配分を少なくすることによって部品の環境温度を広げるなどの方法を採用して対応するため部品コストを押し上げたり、信頼度を下げたりする要因となって現れてくる。すなわち、ミッション重量を多くしたり、姿勢の制御精度を上げたりしようするとコストや信頼性に影響することを意味している。図-23 は、このような関係を示しており、図-24 は一般的な相関を示している。もちろん人工衛星は、開発に5～7年を有するため、開発期間もこの関係に無縁ではない。このような開発に当たっての要素間の関係をバランスをとったり、調整して要求条件を設定することをトレードオフと呼び、これを開発の初期、すなわち企画／計画段階で行うことが重要

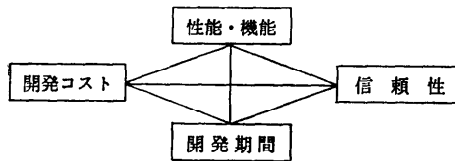


図-23 開発中のトレードオフの関係

である。

そこで人工衛星で用いる部品、設計条件などのデータベースがある程度設定できていれば、信頼度をパラメータとして図-24 の関係をコンピュータを通じて計算し、他の政策的な条件を受けながら機器設計を行い、部品の選定条件、動作温度条件、冗長設計などを行うことができる。この計算のためのプログラムを「電子機器の信頼性設計用プログラム」(略称 SYROP*)と呼んでおり、昭和 57 年から 60 年にかけて宇宙開発

* System Reliability Optimization Program.

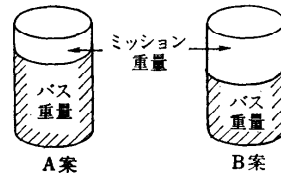
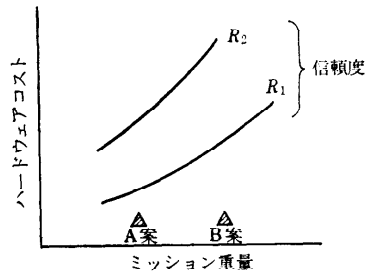


図-24 人工衛星の重量とコストの関係

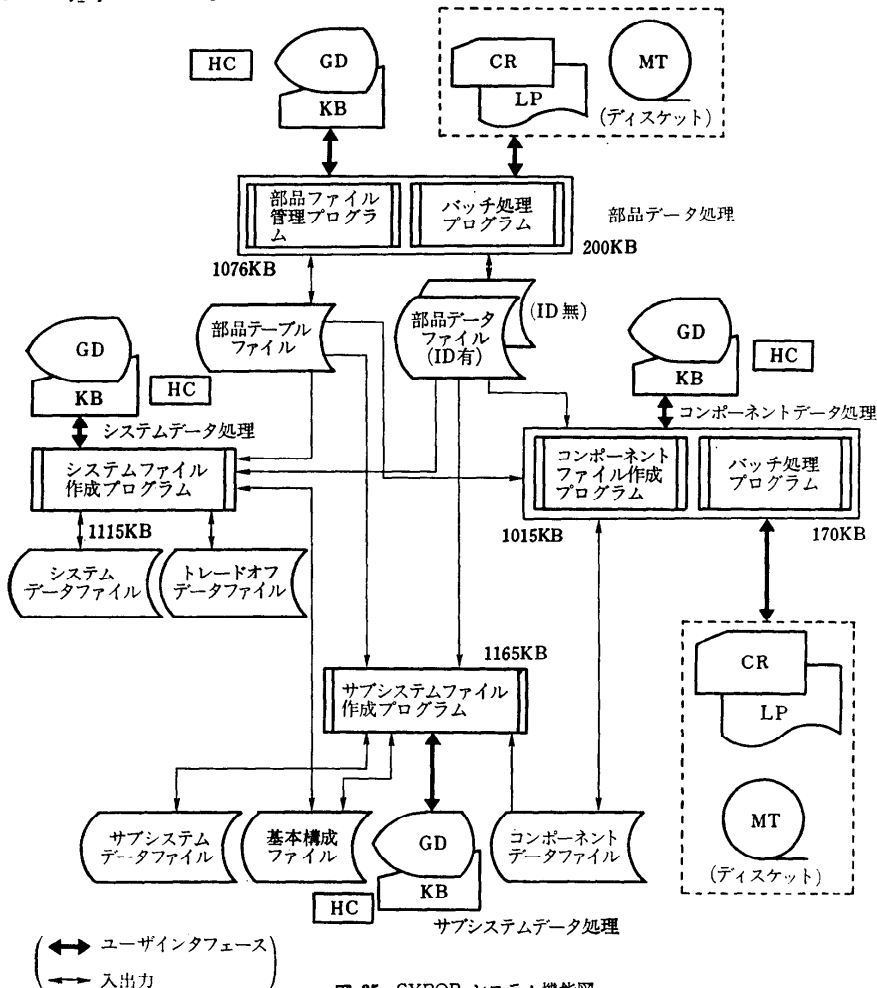


図-25 SYROP システム機能図

事業団で開発した。62年度現在このプログラムは可動しているが、データベースの最新化、設計条件の拡張などを考慮し、次世代用プログラムを開発すべく準備中である。以下に現用の SYROP の概要を紹介する。

(1) SYROP のシステム機能を図-25 に示す。データファイルとして部品データファイルなど7種類、ファイル作成プログラムとして部品管理プログラムなど6種類を用意してある。このプログラムは対話形式をとり、部品、コンポーネントのデータを外部記憶装置から連続的に読み込み、書き出しはバッチ処理による。対話処理はグラフィックディスプレイ T-4114 によって行う。バッチ処理はワークステーション端末により行う。各プログラムはモジュール構造を採用し、言語は、FORTRAN 77 を基本とした。そし

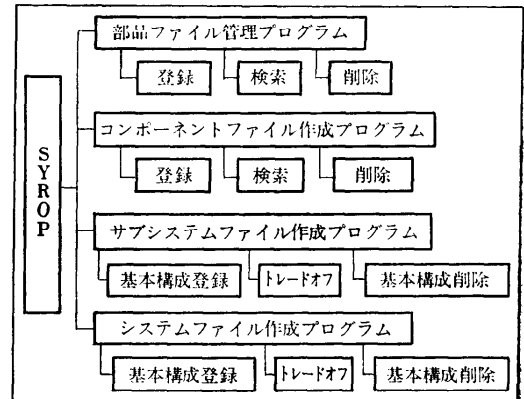


図-26 対話処理プログラム

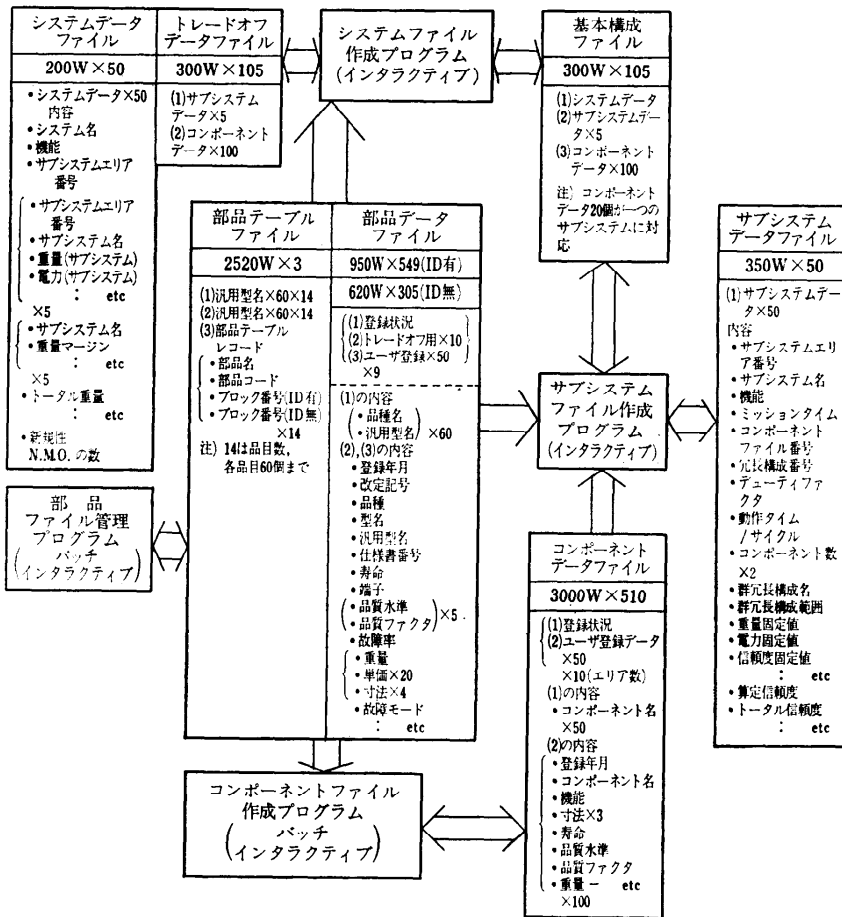


図-27 ファイル構成

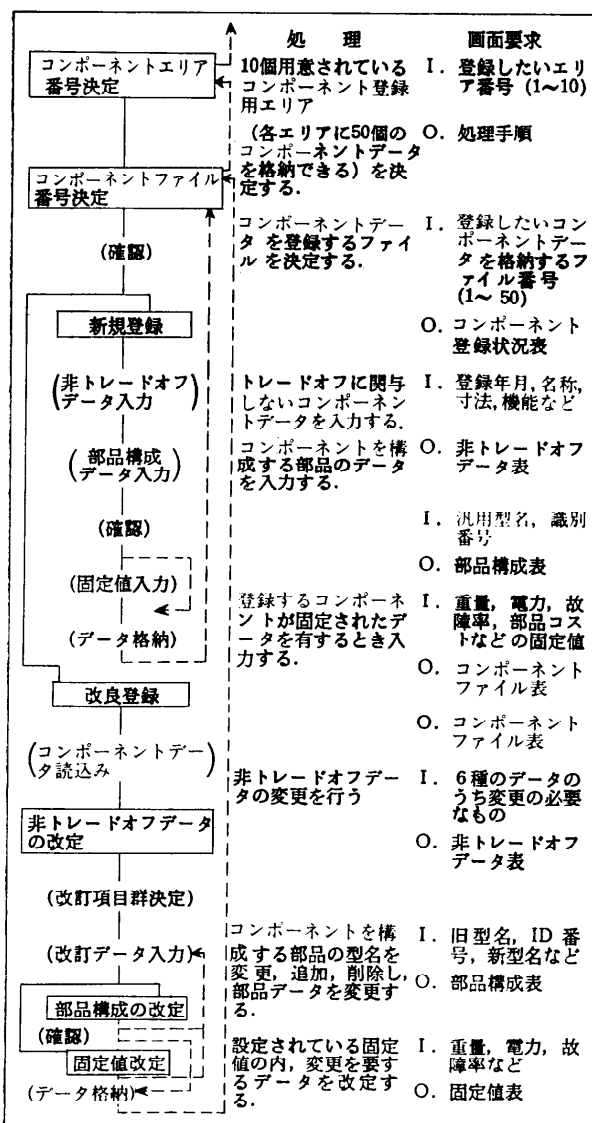


図-28 登録処理フロー

てユーザは特定の信頼性技術者を対象とせず、人工衛星の電子、電気設計技術者が使うことを想定した使いやすい構成となっている。

(2) SYROP の中心である対話処理のために4本のプログラムがあり、図-26に示す。ファイルの構成を図-27に示す。図-27の矢印はデータの流れを示している。図から明らかのように部品データファイル、部品テーブルファイルは、SYROP の各プログラムに共通に使用され、データバンク的構成の源となっている。図-27のうち、コンポーネントファイル作成プログラムを例にとって説明する。

L/I	名 称 と 算 式
11	シングル $R = \exp(-\lambda \cdot t)$
12	シングル/デューティファクタ $R = \exp(-k \cdot \lambda \cdot t)$ $k = D + 0.1(1-D)$, D ; duty factor
13	シングル (オペレーションタイム・サイクル) $R = \exp(-\lambda \cdot t_{op})$ or $R = \exp(-\lambda \cdot t_{cycle})$
21	パラレル $R = 2RR - RR^2$ $RR = \exp(-\lambda \cdot t)$
22	パラレル/デューティファクタ $R = 2RR - RR^2$ $RR = \exp(-k \cdot \lambda \cdot t)$ $k = D + 0.1(1-D)$ D ; duty factor
23	パラレル/オペレーションタイム or サイクル $R = 2RR - RR^2$ $RR = \exp(-\lambda \cdot t_{op})$ $RR = \exp(-\lambda \cdot t_{cycle})$
31	スタンドバイ $R = RA(1 + 10(1 - RB))$ $RA = \exp(-\lambda \cdot t)$ $RB = \exp(-0.1 \cdot \lambda \cdot t)$
32	スタンドバイ/デューティファクタ $R = RA(1 + 10(1 - RB))$ $RA = \exp(-k \cdot \lambda \cdot t)$ $RB = \exp(-0.1 \cdot \lambda \cdot t)$ $k = D + 0.1(1 - D)$ D ; duty factor
41	A out of B $R = \sum_{i=A}^B c_i$ $RR^i(1 - RR)^{B-i}$ $RR = \exp(-\lambda \cdot t)$
42	A out of B スタンドバイ $R = R_1 + \sum_{i=1}^M A_i R_i (1 - R_1)^i$ $R_1 = \exp(-(A + 0.1M) \cdot \lambda \cdot t)$ $R_2 = \exp(-(A + 0.1(M-i) \cdot \lambda \cdot t)$ $R_3 = \exp(0.1 \cdot \lambda \cdot t)$ $M = B - A$ $A_i = \frac{(A + 0.1M) \cdots (A + 0.1(M-i+1))}{i! \cdot (0.1)^i}$
51	パラレル/パラレル $R = R_1 + R_2 - R_3$ $R_1 = \exp(-\lambda_1 \cdot t)$ $R_2 = \exp(-\lambda_2 \cdot t)$ $R_3 = \exp(-(\lambda_1 + \lambda_2) \cdot t)$
61	マルチ $R = R_1 \cdot R_2 (3 - 3R_1 R_2 + R_1 R_2^2)$ $R_1 = \exp(-\lambda_1 \cdot t)$ $R_2 = \exp(-\lambda_2 \cdot t)$

図-29 システム構成

コンポーネントファイル作成プログラムは、重量、消費電力、故障率、部品コスト、部品品種数などのトレードオフデータの算出、コンポーネントデータファイルの維持管理を行う。大別すると登録、検索、削除の3機能がある。コンポーネントデータとして入力するものには、非トレードオフデータ (登録年月日、名称、機能、寸法、寿命など)、構成部品データ (型名、識別番号、個数、品質水準)、さらにトレードオフデータのうち、値を設定しておくべき固定値データに3種類があり、ディスプレイにおける対話を通じて入力し、徐々にコンポーネントの設計を行っていく。ディスク

上のデータを入力してコンポーネントトレードオフ表、同ファイル表、部品構成表をディスプレイ上に出力していく。決定した構成部品は、非トレードオフデータや固定値とともにファイルをするがその処理の流れを図-28に示す。

(3) このプログラムで用いるアルゴリズムは、きわめて取扱いの多いものを採用し、コンポーネントの各種パラメータ算出のために次のものをもつ。ここで N は 200 以下としてある。

- (a) 重量: $\sum_{i=1}^N (\text{部品重量} \times \text{部品個数}) + \text{固定値}$
 (b) 消費電力: $\sum_{i=1}^N (\text{部品消費電力} \times \text{部品個数}) + \text{固定値}$
 (c) 部品コスト: 単価 + ロット費 + 固定値
 単価: $\sum_{i=1}^N (\text{部品単価} \times \text{部品個数})$
 ロット費: $\sum_{i=1}^N (\text{部品ロット費用} \times \text{品質ファクタ}^*)$
 (d) 部品数: $\sum_{i=1}^N \text{部品個数}$

(e) 品種数: $\sum_{i=1}^N (\text{部品型名})$

(f) 故障率: $\sum_{i=1}^N (\text{部品故障率} \times \text{部品個数} \times \text{品質ファクタ}^{**})$

人工衛星の設計では、冗長構成はコンポーネントとして採用され、サブシステムとしては採用されない。そのため SYROP では、始めから一般的に採用される冗長構成を 12 通りもっており、設計者はそれから選定する。図-29 にその 12 通りのシステム構成を示す。そして求めるデータは次のものとなる。

- (a) 重量
 (b) 消費電力
 (c) 部品費
 (d) 部品数
 (e) コスト
 (f) 信頼度

* コストを決定するためにとられた品質保証レベル。

** 故障率を決定するためにとられた品質保証レベル。

<<< DISPLAY OF SUBSYSTEM FILE >>>

SUBSYSTEM FILE

FILE NO. 121

SUBSYSTEM AREA : 3
 SUBSYSTEM NAME : AFF. TRANSFER MODE 4
 MISSION TIME (HOUR) : 100
 FUNCTION : TESA, SUN SENSOR (IRU2)

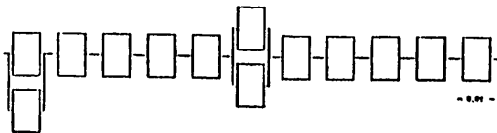
WEIGHT (KG) : 31.0 (31.0 + 0.0 (FV))
 POWER(W) : 47.5 (47.5 + 0.0 (FV))
 RELIABILITY : 0.99320 (0.99320 + 1.00000 (FV))

PARTS COST (1000YEN) : 1207000 (1207000 + 0.0 (FV))
 COST (1000YEN) : 1207000 (1207000 + 0.0 (FV))
 PARTS COUNT : 8
 COMPONENT COUNT (NEW) : 8
 COMPONENT COUNT (MODIFIED) : 0
 COMPONENT COUNT (OFF-THE-SHELF) : 0

NOTES

VOICE, RIBAT, FDEP1, VDE, IRU2=TEIJ/O MODE3
 CSSE=ACQUISITION MODE4
 FUSION X1, CSS4 X6, TESA X2 GA TUIKA

426 433 443 417 447 427 436 437 437 437 421



1 2 3 4 5 6 7 8 9 10 11
 6 P 8

DO YOU WANT TO FINISH YOUR RETRIEVAL
 (Y OR N)

Y : RETURN TO SORTING KEY INPUT PROCESS
 N : RETURN TO DISPLAY OF SUBSYSTEM TRACE-OFF FILE
 ???

図-30 ① SYROP 画面の例

SYSTEM FILE 16
SYSTEM-OFF FILE 2

<<< DISPLAY OF SYSTEM TRADE-OFF FILE >>>

SYSTEM TRADE-OFF FILE

NO.	SYSTEM NAME	WEIGHT (KG)	POWER (W)	RELIABILITY	P-COST (*1000YEN)	P-COUNT	COMPONENT N H O	COST (*1000YEN)
1	ETS-VI ACS 2	87.6	147.0	0.669116	1520000	0	91 0 12	1620000
2	ETS-VI ACS 1	68.4	147.0	0.640734	1602000	0	93 4 21	1697000
3	ETS-VI ACS 6	79.4	157.0	0.645300	1710000	0	93 10 0	1710000
4	ETS-VI ACS 5	76.2	157.0	0.645870	1702000	0	89 7 14	1700000
5	ETS-VI ACS 4	76.6	151.0	0.603947	1670000	0	02 0 12	1070000
6	ETS-VI ACS 8	77.4	143.0	0.742176	1710000	0	91 0 12	1710000
7	ETS-VI ACS 3	79.4	151.0	0.644262	1652000	0	91 4 20	1650000
8	ETS-VI ACS 7	70.2	143.0	0.742131	1702000	0	90 4 10	1700000
9	ETS-VI ACS 12	80.5	150.0	0.730100	1702000	0	97 4 7	1700000
10	ETS-VI ACS 11	82.8	150.0	0.750100	2002000	0	93 4 14	2000000
11	ETS-VI ACS 13	102.5	157.0	0.679141	2072000	0	90 4 7	2070000
12	ETS-VI ACS 9	102.8	154.0	0.725600	2352000	0	91 4 14	2350000
13	ETS-VI ACS 10	112.8	158.0	0.720753	2702000	0	95 4 14	2700000
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								

PAGE 1

```

+-----+-----+-----+-----+-----+-----+-----+-----+
| INPUT NO. | PROCESS |
| 0 | END(RETURN TO SORTING KEY INPUT PROCESS) |
| "CH" | NEXT PAGE |
| FILE NO. | DISPLAY OF SYSTEM FILE |
|??? PLEASE INPUT A FILE NUMBER |

```

図-30② SYROP 画面の例

(4) SYROP は独立した画面として 120 面もっている。各ファイルごとにいろいろ工夫しているが、その代表的なものの 2 面を図-30①、②に示す。

(5) このプログラムによる信頼度の計算は、単なる信頼度を求めるためのものではなく、人工衛星の開発初期段階特に企画、計画段階、遅くともシステム定義段階における人工衛星の開発計画策定のために用いることを目的としている。一般に人工衛星は、その目的とする機能が初めに決められ、その実現性が続いて検討される。ミッション重量は、その目的機能を実現するために、おおよそその値が推定されるが、それを受け入れられるかどうかは、バス機器側の設計作業と平行して検討される。その時点ですでにコスト、信頼度とともに情報として提供されなければ、有効な開発計画は設定できない。すなわち、機能だけが先行して決定されると、後段階の開発に無理が生じ、結果としてバランスのとれた人工衛星が実現できないことを示唆している。その意味からこのプログラムの存在は大きい。もちろんまだこのプログラムは初期的なもので完

壁な情報を与えるものではないが、部品の標準化によるコストの低減化、重要な部分への重点投資、冗長系の有効性の検討など、本来経験のみに頼っていた設計作業が、具体的な表示、それに続くトレードオフを通じて人工衛星のシステム仕様書が有効に設定できるようになると期待している。

6. ま と め

大規模システムの一つの例である人工衛星に関する信頼性技術について紹介した。内容をまとめると次の二つに分類できる。

(a) 人工衛星の開発に当たって、経済的で確実なシステムを確立するために、信頼性技術、特に信頼度を用いたトレードオフ及び設計支援業務が必須である。

(b) 最終的には無欠陥の衛星を指向するために、試験、解析、管理などの業務を進める。特に経験情報の十分な活用と故障解析は必須である。

またシステム内にはコンピュータが採用されてきてお

り、このソフトウェアの品質保証は、人工衛星システムのきわめて重要な要素である。そして最終的には、このような信頼性技術が有効に活用できるためには、管理組織と手順、特にコンフィギュレーション管理が有効に働くことが必要であることを付言してまとめる。なおソフトウェアの品質保証についてまとめるに当たり、当事業団鈴木孝氏に助言をいただいたことに対し謝意を表する。

参 考 文 献

- 1) 市田 嵩, 下平勝幸: 信頼性管理, 日科技連信頼性工学シリーズ 15, 日科技連 (昭和 59 年).
- 2) 宇宙開発事業団: 信頼性プログラム共通仕様書, NASDA-SPC-1177 (昭和 51 年).
- 3) 下平勝幸, 新田晃道, 稲川美之, 春井勝一: SYROP—開発報告と適用実施例—, 電子通信学会信頼性研究会資料, R 85-13 (昭和 60 年).

(昭和 62 年 6 月 22 日受付)