

情報フロー・ダイアグラムによる 組込みソフトウェア非正常系の分析手法

亀谷 秀洋[†] 新屋敷 泰史[‡] 三瀬 敏郎[‡] 橋本 正明[†]
鵜林 尚靖[†] 片峯 恵一[†] 中谷 多哉子[§]

[†]九州工業大学 〒820-8502 福岡県飯塚市川津 680-4

[‡]松下電工株式会社/九州工業大学 〒571-8686 大阪府門真市大字門真 1048

[§]有限会社 エス・ラグーン/九州工業大学 〒272-0143 千葉県市川市相之川 4-12-14

E-mail: [†]{kametani,hasimoto}@minnie.ai.kyutech.ac.jp

あらまし 本稿は、組込みシステムとその動作環境の情報フローをダイアグラムに表現し、そのダイアグラムを用いて組込みソフトウェアの非正常系分析を行うための手法を提案する。近年の組込みシステムは、大規模かつ複雑化、さらに製品開発サイクル短期の要求が著しい。このような現状においては品質問題が発生しやすくなり、製品のリコールは企業にとって大きな問題となる。高品質な組込みソフトウェアを開発するには、非正常系仕様をシステムの分析・設計時に十分に検討しておくことが重要である。

キーワード 組込みソフトウェア, 非正常系, 情報フロー・ダイアグラム, 分析

An analysis method of unexpected obstacles in embedded software by Information Flow Diagram

Hidehiro KAMETANI[†] Yasuhumi SHINYASHIKI[‡] Toshiro MISE[‡] Masaaki HASHIMOTO[†]
Naoyasu UBAYASHI[†] Keiichi KATAMINE[†] Takako NAKATANI[§]

[†]Kyusyu Institute of Technology 680-4 Kawazu, Iizuka-shi, Fukuoka, 820-8502 Japan

[‡]Matsushita Electric Works, Co., Ltd./Kyusyu Institute of Technology 1048 kadoma, kadoma-shi, Osaka, 571-8686 Japan

[§]S-Lagoon, Ltd./ Kyusyu Institute of Technology 4-12-14-102 Ainokawa Ichikawa-shi, Chiba, 272-0143 Japan

E-mail: [†]{kametani,hasimoto}@minnie.ai.kyutech.ac.jp

Abstract This paper proposes an analysis method of unexpected obstacles in embedded software by using diagrams on which information flow in embedded systems and their operational environment is drawn. Recently, embedded systems have been becoming large in scale and complicated. Moreover, the development cycle of the systems is required to be shortened. This trend of industry troubles the quality of products. Recall of products is one of the serious problems for companies. For improving the quality of embedded software, it is indispensable to analyze unexpected obstacles in the process of requirement analysis and design.

Keyword Embedded Software, Unexpected Obstacles, Information Flow Diagram, Analysis

1 はじめに

近年、組込みソフトウェアは、高機能化や複雑化に加え、開発期間の短縮に対するニーズが強まっており、品質要件の確保が重要な課題となっている。組込み製品は、ソフトウェアを意識しない一般不特定ユーザーにより、不特定環境下でそれぞれの用途に合わせて運用され、安全・安心・長寿命な動作が期待されている。そのような品質確保を行うため、ソフトウェアの70%強は通信エラー対応や、信号のノイズ除去、誤操作対応、障害回避機能といったエラー処理のソフトウェアである。しかし、そのような非正常系の仕様はハ

ードウェアの知識と、システムの動作環境の知識が必要となるため、ソフトウェア開発者にとって見落としやすい。実際に、そのような仕様の抜けが、開発見積もりの誤り、設計やり直しによる開発遅れ、あるいは、テスト仕様の抜けによる製品の品質や安全の問題となっている。

そのため、非正常系に対応するための仕様を、正確かつ効率的に設定できれば、組込みソフトウェア開発における大きな改善効果が期待できる。そこで、我々は組込みシステムにおける異常や例外など、要求定義や設計の段階において考慮から漏れ易い事項を非正常

系と呼び、その仕様を設定するための組込みソフトウェア非正常系の分析手法を研究している。本稿では、正常系および非正常系を次のように定義する。

正常系：ソフトウェア使用マニュアルに記述されているような、設計時に予期でき、定義される振る舞い

非正常系：ハードウェアの故障や異常動作、および劣化、動作環境における、オーバーロード、誤操作、誤使用、自然環境における、高温、信号ノイズ、雨、など、正常系からの逸脱

本研究の目的は、これまで経験豊富な開発者によってなされてきた仕様の妥当性確認手法を形式化することにより、非正常系仕様の欠落を抑止することである。本分析手法では、組込みシステムにおける異常や例外の発生を、システム内の伝達すべき情報が本来意図したものとは違う情報へ変化した事と捉えている。そこで、本稿は、動作環境も含めたシステム内の情報フローをダイアグラムに表現する事によって、情報変化の発生可能性を可視化し、さらに、そのダイアグラムを分析することにより、非正常系発生の原因と、それがシステムに与える影響を分析するための手法を提案する。

本稿の第2章では、組込みシステムのモデリングに関する概念を述べる。第3章では情報フロー・ダイアグラムを提案し、第4章で情報フロー・ダイアグラムを用いた分析手法を述べる。第5章では事例分析を行い、第6章で考察する。

2 組込みシステムのモデリング

組込みシステムのモデリングの概念を述べる。

2.1 組込みシステムの捉え方

業務系システムでは、CPUや、メモリ、ハードディスク等かなり限定されたハードウェア構成の上に、ソフトウェアでシステムの振る舞いを決定する。一方、組込みシステムは製品ごとに様々なハードウェアの構成が存在する。そのため、それぞれのデバイスに着目することが重要となる。したがって、組み込みシステムを、ユーザや動作環境も含めてモデリングを行うことが有効である。

組込みシステムは、主にユーザの操作や動作環境から来る信号によって動作する。すなわち、ユーザや動作環境の要素がシステムのアクターとなり、そこからシステムに対して、トリガとなる信号が発せられる。アクターから発せられた信号は、各デバイスを通り、マイコン内のソフトウェアで認識され、処理される。ソフトウェア処理された後は、各デバイスに意図通りに動作するように信号が送られ、システムとして動作する。

ここで、システムの誤動作について述べる。正常系アクターはシステムに対して要求を発する。システムはその要求を正確に解釈し、満たすように振舞うことによって製品としての役割を果たす。一方、アクターの要求を誤って解釈してしまうことや、要求に反する振る舞いを行うことはシステムの誤動作である。また、誤ったアクターから来る信号に対してアクションを起こすことも、システムの誤動作である。システム側から見れば、ある情報を受け取り、解釈し、処理を行い、

また情報を出力するという過程において、何らかの原因によって、正式な情報が誤った情報に変化する。すなわち、システムの誤動作の原因は、システム内あるいは動作環境において誤った情報が伝達されるところにある。

また、システムの誤動作により、障害を引き起こす場合がある。この障害発生も情報の誤伝達により、引き起こされると捉えている。何らかの要因によって、情報の誤伝達が起こる。さらにその誤った情報が伝達されて、他のデバイスで解釈され、処理され、出力され伝達される。このように、障害の発生は、ある要因から誤情報が伝達され、その誤情報が次々に伝達されるというシナリオを持つ。このシナリオを障害シナリオと呼ぶ。

本研究では、システムと動作環境の中を伝達される情報のフローに着目することが、非正常系分析の支援において効果的であると考えている。そこで、システム内の各デバイスによって伝達される情報と、システムと交信する動作環境の中を伝達される情報について、そのフローを、ダイアグラムを用いて視覚化する。また、ダイアグラム上で情報フローを分析し、障害シナリオを抽出することによって、非正常系を分析する。

2.2 情報伝達の概念

本稿の組込みシステムにおける情報伝達の概念を述べる。我々は、組込みシステムを情報の伝達により動作していると捉えている。ある電氣的デバイスの間の接続に着目した場合、そこでは電気信号が伝わることによって動作している。実際には、その電気信号は単に電圧や周波数といった物理的パラメータを運んでいるだけでなく、システムが意思決定するための情報を運んでいる。例えば、モータ ON の信号やセンシングした値などの情報を含んでいる。すなわち、デバイス間を流れる電気信号や、動作環境と組込みシステムの間を流れる光や温度などの物理的媒体は、情報を運んでいる。この情報の運び手をキャリアと呼ぶ。したがって、システムにとって意思決定に必要な情報は、キャリアによって伝達されている。情報伝達の概念を図1に示す。

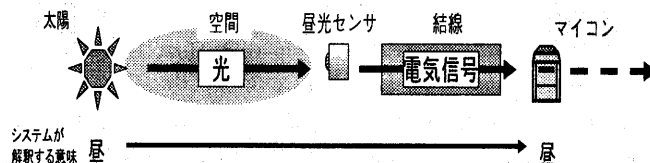


図1 情報伝達の概念

3 情報フロー・ダイアグラム

本研究で考案した情報フロー・ダイアグラム（以下 IFD (Information Flow Diagram) と呼ぶ）について述べる。

3.1 デバイス・ダイアグラム

デバイス・ダイアグラムにおいては、組込みシステム中のデバイスや、動作環境に含まれるユーザなどのオブジェクトをモデリング対象ドメインに含む。なお、対象ドメインには実際にシステムを構成するデバイスのインスタンスを記述する。

各デバイスには、その名前と属性、および、そのデ

バイスで起きる可能性がある故障のパターンを明記する。故障パターンは、ノイズ、断線、劣化などの故障の種類を示す。属性は、分析に必要なデバイスの性質を表すパラメータである。図2にデバイスの表現を示す。

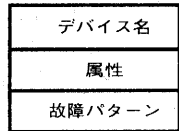


図2 デバイス

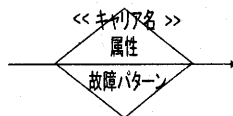


図3 キャリア

3.2 プロセス・ダイアグラム

IDEF0 (Integrated Definition methods) はアクティビティ・モデリング手法とも呼ばれ、企業などの活動を機能的な側面から体系付けた静的機能モデルを構築するための手法である。IDEF0 では、業務プロセスをアクティビティという単位に分割し、それぞれのアクティビティに対して、入力と出力、およびコントロールとメカニズムを指定する。アクティビティは、入力情報を基に、メカニズムによって処理を行い、出力情報を生成する。メカニズムの処理を行う際の条件が、コントロールである。IDEF0 プロセスの表現を図4に示す。本稿では、この IDEF0 をベースに、組込みシステムのモデリングを行う。

プロセス・ダイアグラムは、各デバイス上で処理されるプロセスと、そのプロセス間の情報フローと制御フローを表す。作成方法は、IDEF0 に従ってデバイス・

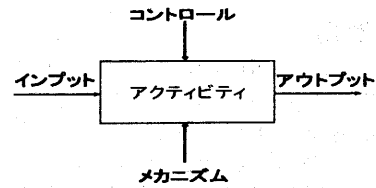


図4 IDEF0 のプロセス

ダイアグラムに記述されたデバイス上で処理されるプロセスを記述し、その入力と出力、コントロール、および、メカニズムを記述する。入力と出力、コントロールには、流れる情報の名称を付記し、メカニズムとしてはデバイス・ダイアグラムからそれぞれのプロセスにアローを繋ぐ。

IDEF0 は段階的に詳細化して記述可能なため、初めは最上位階層のプロセスのみ記述する。その後、必要に応じてプロセス毎に下位の階層として詳細に記述する。正常系の IFD の例を図5に示す。

4 非正常系分析手法

IFD を用いて非正常系分析を行う手法を述べる。

4.1 FMEA の適用

IFD 上で FMEA (Failure Modes and Effect Analysis) を用いて非正常系を分析する手法を述べる。IFD 中のデバイス・ダイアグラムのデバイスに着目する。そのデバイスの故障パターンを1つピックアップし、その故障パターンが発生した場合の情報フローを実際に IFD 上に表す。デバイス故障が、システムに与える影響を分析する。故障パターンの影響がソフトウェアで正常に処理される場合は問題無い。一方、正常に処理されずに障害を引き起こす場合は、IFD 上で、故障パターンから障害に至るまでの情報フローが存在する。すなわち、障害シナリオが抽出される。このように、FMEA を適用することによって、デバイスで故障が発生した場合のシステムの振る振る舞いを分析して、障害シナリオを抽出することができる。

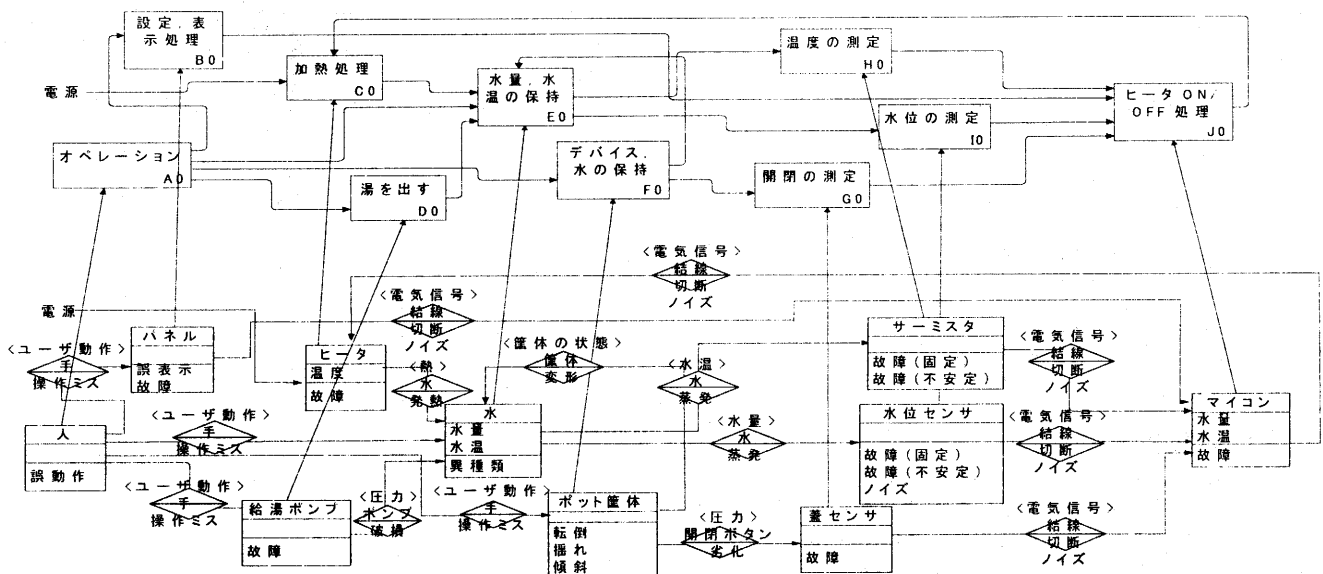


図5 情報フロー・ダイアグラム

4.2 FTA の適用

障害とその障害を引き起こすキャリアを品質項目に従って整理したものを、障害リストとして作成する。障害リストの例を表 1 に示す。

表 1 障害リストの例

品質項目	障害	キャリア
安全性	火災	熱
	火傷	熱

...

FTA (Fault Tree Analysis) 適用の目的は、障害リスト中の障害につながる非正常系を洗い出し、障害シナリオを抽出することにある。その分析手法を以下に述べる。

まず、障害リスト中の障害を引き起こすキャリアを 1 つピックアップする。そのキャリアが存在するパスを IFD 上から探し出す。そのパス上で該当障害が発生する状況进行分析し、IFD に記述する。以下これを繰り返し、原因を見つけ出す。この原因はデバイスの何らかの故障パターンに集約され、この原因から障害までの情報フローが、障害シナリオとなる。

4.3 ガイドワードの適用

IFD においてガイドワードを用いた分析手法を説明する。HAZOP (the Hazard and Operability) は元々、化学プラントの危険要因を分析する手法である。HAZOP におけるガイドワードを組込みシステムの分析にも適用する。化学プラントでは化学物質の流れがガイドワードの対象であるが、組込みシステムでは情報の流れとキャリアを対象とする。IFD 中の正常系情報フローに対し、表 2 に示すガイドワードを適用し、その影響进行分析する。

ガイドワードは想定事項なので、ガイドワードが表す逸脱は、必ず発生するとは限らない。また、ガイドワードの逸脱がシステムに対して障害を引き起こすとも限らない。そのため、ガイドワードの逸脱の発生原因と、その逸脱によって引き起こされる障害が存在することは、障害シナリオの抽出に役立つ。

表 2 ガイドワード

情報の意味	大きくずれた(値)/小さくずれた(値) 状況から矛盾した(値)
情報の量	多すぎる/少なすぎる 長すぎる/短すぎる
情報の時間	タイミングが早すぎる/遅すぎる 同期してしまう/同期がはずれてしまう 長すぎる/短すぎる
情報の構成	順序が異なる 構成が異なる
情報の対象	本来の対象ではない
情報の状態	喪失した、不定な、固定してしまった 不安定な、発振した

4.4 体系的分析手法

IFD を用いた非正常系分析手法を説明する。まず、正常系の IFD を作成し、正常系情報フローに対してガイドワードを適用する。本来伝達されうる正常情報がガイドワードに従って逸脱した場合を想定することによって、逸脱した情報がシステムにどのような影響を

与えるか分析する。その逸脱情報は最終的に障害に至る場合と正常に処理される場合とがある。その中で障害に至る逸脱の場合、逸脱する原因を分析する。ブレイクダウンしていく中で、デバイスやパスの故障パターンを発見できれば、障害が発生する可能性が存在することになる。一方、原因となる故障パターンが存在しなければ、ガイドワードの逸脱そのものが発生しないので、その逸脱に対する処理を追加する必要はない。故障パターンからガイドワードを通過し障害に至る経路において、その経路上の一連の情報の流れを追うことは、障害に至るシナリオを表現している。この障害シナリオが分析の成果物となる。

また、非正常系は思いつきにくいいため、FMEA 適用時には障害自体が見落とされやすく、FTA 適用時には障害の発生原因が見落とされやすい。FMEA と FTA は分析の方向が異なるが、分析結果の障害シナリオにおける、故障パターンから障害に至る情報フローは変わらない。すなわち、FMEA と FTA は分析過程のある点においてつながる。FMEA と FTA を単独で適用するのではなく、ガイドワードとも合わせて適用することによって、個々の適用では双方で見落とされやすい現象を補え、発見しにくい非正常系を抽出できる。

5 事例分析

本分析手法の適用事例を示す。

5.1 適用事例

事例として、話題沸騰ポット要求仕様書[1]を用いる。本ポットは、温度センサや、5 段階の水位センサ（最下位は水切れ検出、最上位は満水検出）、蓋空き検知センサ、ヒータ制御、マイコン、ポンプ、表示パネルを持つ。沸騰ボタンを押すとポット内の水を沸騰させ、カルキ抜きを行なった後、保温になる。給湯ボタンを押すと、ポンプにより給湯を行なう。蓋が開いているときは、ヒータを ON し水を加熱することは行なわない。

5.2 事例分析

ポットは、水を沸騰している途中に、蒸発などによって水が無くなると、ヒータを止めるように働く。この機能の実装方法として、水位センサによって水位を検知しておき、ポット内の水が空になったら、つまり第 1 水位センサで水を検知しなくなったら、止めるということが考えられる。本分析手法を用いて、このような実装方法が最善であるか分析する。まず、第 1 水位センサのアウトプットの情報に対して、「矛盾した」というガイドワードを適用する。ポット内の水が空の場合、各水位センサは「水は無い」という情報をアウトプットするが、ガイドワード適用により、「矛盾した」情報である、「水がある」という情報をアウトプットすると想定する。これを図 6 に示す。

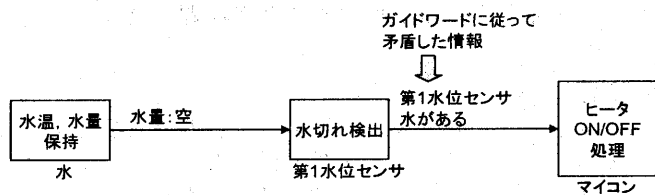


図6 ガイドワードの適用

次に、この状況からどのような障害が発生するか分析する。マイコンにおいては水位センサから伝達されてきた情報に基づいてポット内に水が存在すると判断し、沸騰を続ける。しかし、実際にはポット内には水は存在しないので、ヒータが異常加熱し、熱が筐体へと伝わり、筐体から発火してしまう恐れが判明する。これを図7に示す。

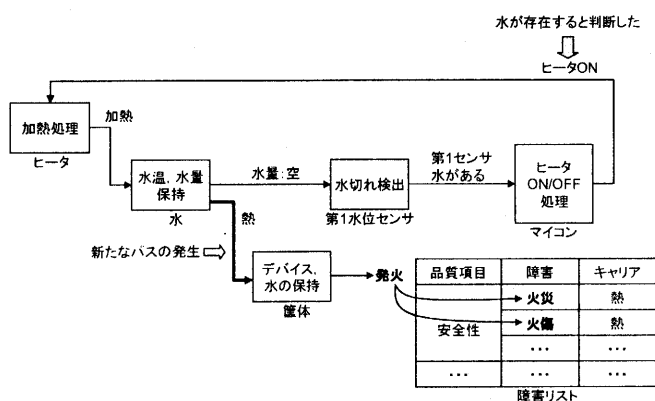


図7 発火までの情報フロー

次に、第1水位センサがガイドワードのように「矛盾した」情報を伝達する場合は存在するかどうか分析する。デバイス・ダイアグラムの第1水位センサを見ると、図8のように、故障パターンとして故障（固定）が存在する。故障（固定）とは、入力値に関係なく、固定値を出力し続ける故障パターンである。したがって、たとえ水が無いことを検知していたとしても「水がある」という情報を出し続ける固定の故障が起こってしまう可能性がある。

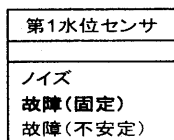


図8 第1水位センサ

このことから、第1水位センサにおいてガイドワード「矛盾した」に至る故障パターンが存在することが分かる。以上から、次のような障害シナリオが得られる。加熱沸騰途中にポットの水が空になった時、通常ヒータの電源を落とさなければならないが、第1水位センサにおいて故障（固定）が発生している場合、マイコンではポット内に水が存在すると判断し、加熱し続け、ポットが発火する。分析結果であるこの障害シナリオから、加熱する際には、ポット内の水の有無を水位センサのみで判断するだけでは不十分であるという結論に至る。この対策として、水位と温度変化から

ポット内の水の量を推定し、ヒータの制御を行うなどの方法が考えられる。

5.3 故障パターンの組合せ

FMEAにおいては基本的に、デバイスの故障パターンの組合せは考えられていない。しかし、高品質化のための非正常系分析を行なうには、故障パターンの組合せによってシステム全体がどのような影響を受けるか分析しなければならない。そのため、IFD上でデバイス故障パターンの組合せの影響も分析する。しかし、故障パターンの組合せを取る場合に、単純に全ての組合せを取ると組合せの数が爆発する。そこで、故障パターンを組合せる手法を述べる。

各デバイスはインプットに対して処理を行い、アウトプットを出す。すなわち、1つのデバイスで行う処理を見た場合、他のデバイスでどのような障害が発生していようが、インプットの情報のみに影響を受ける。そのため、デバイス障害の組合せを取ることは、各デバイスにおいてインプットの情報に対して、その情報を出力するデバイスの障害との組合せを取れば良い。このように、単純にデバイス障害の組合せを取るのではなく、組合せ爆発を起こさずに、障害の組合せを分析することができる。そのためには、各デバイスにおいて、入力情報を抜き出さなければならない。次に、入力情報を漏れなく列挙する手法を述べる。

各デバイスが行なう処理は次のように分類することができる。

1. インプットに対して処理を行い、アウトプットを出力する。
2. アウトプットのみが存在する。
3. インプットのみが存在する。

各デバイスを流れる情報を列挙するには、まずシステムの情報の発信源から始める。上記の2に当たるプロセスが情報発信源に当たり、ユーザや動作環境の要素がこれになる場合が多い。情報発信プロセスが特定できた後、このプロセスから他のプロセスへ出力される情報を全て洗い出す。この際、正常系と非正常系ともに列挙する。次に、情報発信プロセスから出力された情報を入力するプロセスにおいて、全ての入力される情報と、故障パターンの組合せを含めた処理との総組合せを取り、出力を列挙する。以下、下流プロセスでもこれを繰り返し行なうことによって、全てのデバイスの障害の組合せを含めた情報フローを求めることができる。

デバイスの故障パターンには、様々な種類がある。たとえば、全く動作しなくなるものもあれば、レベルが下がるといった劣化も存在する。これらの故障パターンの種類を分けることによって、故障パターンの組合せを考える際に、無駄な組合せを行わなくて済む。あるデバイスにおいて、全く動作しない故障パターンが起こっている時、そのデバイスで他の故障パターンがいくら起こっても、結局動作はしない。つまり、同一デバイスにおいて、全く動作しない故障パターンが発生している時は、他の故障パターンの組合せを考える必要は無い。このように、故障パターンの重さに順位を付けることによって、組合せ数を減らせることができる。

6 考察

情報フロー・ダイアグラムを用いた非正常系分析について、考察を述べる。

(1) 既存分析手法の組合せ

本稿では非正常系分析として FMEA, FTA, ガイドワードの3つの分析手法を適用することを述べた。これは、非正常系分析には、分析者の経験やスキルが影響するため、非ベテラン者では抜け落ちが発生してしまうことを防ぐためである。そこで、デバイスからのボトムアップ分析（非ゴール指向）と障害からのトップダウン分析（ゴール指向）を組み合わせることによって、網羅的に分析することにより、無目的に発生する非正常系の抜け落ちを減らすことができる。しかし、実際には双方の分析結果が等しくなる場合が多く、冗長な分析となる可能性がある。

非正常系は無目的に発生し、その影響も様々存在する。そのため、ゴール指向のみによって分析すると、抜け落ちが発生する可能性がある。そこで、ガイドワードや FMEA を用いた手法のように、非ゴール指向により分析することが効果的である。

(2) 情報フロー・ダイアグラムの有効性

非正常系分析を行う際、ベテラン分析者は故障パターンから起こりうる現象を頭の中でつなぐことができる。しかし、非ベテラン分析者では、これを行うことが困難であるため、非正常系の分析に漏れが発生する。

IFD における各デバイスおよびプロセスの障害は、そのデバイスおよびプロセスのアウトプットの情報に含まれる。すなわち、システムを流れる情報中に故障パターンの情報が含まれることになる。そのため、IFD を用いることによって、故障パターンの影響の伝播を追っていくことが可能なため、どのような障害に至るか非ベテラン者でも分析することができる。

また、大規模システムの場合には、システム全体の仕様を把握することが困難である。そのような場合にも、ダイアグラムは理解しやすい。さらに、ダイアグラムを部分的に作成し、結合させることによって全体を表現することができるので、担当部分の知識しかなくても、分析ができると考えられる。

(3) 今後の課題

IFD を用いて分析する際、デバイスの故障パターンと、それが与える影響は知識として必須である。そのため、これらの知識を知識データベース化することによって、ハードウェアの知識を支援するとともに、IFD の作成、あるいは分析を容易にすることができる。今後、このような知識データベースの作成が課題となる。また、情報フローの特性として、フィードバック制御などの情報ループや、情報フローパスの欠落、あるいは新たなパスの発生などがある。IFD をグラフとして扱い、グラフ解析によって非正常系を分析するための手法を確立することも課題である。

我々の研究グループは、組込みソフトウェア非正常系分析を行う手法として、IFD と分析マトリクス[5]を提案している。分析マトリクスを使用した手法では、分析者の経験に頼らなければならない部分があるため、その部分の知識を IFD で補完できると考えている。その際に、双方の知識がどのように関係しているのか解

明し、分析手法を体系化することが課題である。

7 まとめ

本稿では、システムと動作環境を合わせて広義の情報システムとしてモデリングを行った。また、システムと動作環境が交信する情報フローに着目し、IFD を考案した。IFD によりシステムと動作環境の情報が明確になり、見落としやすい非正常系仕様を分析できる。また、既存の分析手法を用いて体系化することで、非正常系の分析を容易にするとともに、分析漏れを少なくすることが可能である。

今後、IFD と分析マトリクスを組み合わせる組込みシステムの非正常系分析手法を体系化することが課題となっている。また、非正常系知識データベースの作成と、IFD のグラフ解析も課題である。

文 献

- [1] 組込みソフトウェア管理者・技術者育成研究会 (SESSAME: Society of Embedded Software Skill Acquisition for Managers and Engineers) “システム教育教材、話題沸騰ポット GOMA-1015 型 要求仕様書”
- [2] 畑中久典, 新屋敷泰史, 三瀬敏朗, 亀谷秀洋, 橋本正明, 鶴林尚靖, 片峯恵一, 中谷多哉子 “組込みソフトウェア非正常系の概念モデルによる情報フロー・グラフの解析”, 電子情報通信学会, 技術研究報告 104-431, pp.19-24, 2004
- [3] 新屋敷泰史, 三瀬敏朗, 江浦洋平, 畑中久典, 橋本正明, 鶴林尚靖, 片峯恵一, 中谷多哉子 “組込みソフトウェア非正常系の概念モデル”, 情報処理学会, 組込みソフトウェアシンポジウム 2004 論文集, pp8-11, 2004
- [4] 三瀬敏朗, 新屋敷泰史, 江浦洋平, 畑中久典, 橋本正明, 鶴林尚靖, 片峯恵一, 中谷多哉子 “組込みソフトウェア仕様抽出のための非正常系分析マトリクス”, 組込みソフトウェアシンポジウム 2004 論文集, pp12-19, 2004
- [5] 三瀬敏郎, 新屋敷泰史, 橋本正明, 鶴林尚靖, 片峯恵一, 中谷多哉子 “組込みソフトウェア非正常系における仕様分析手法の一提案”, 日本ソフトウェア科学会 ソフトウェア工学の基礎研究会 FOSE 2005, pp227-235, 2005.
- [6] T. Mise, Y. Shinyashiki, M. Hashimoto, N. Ubayashi, K. Katamine, and T. Nakatani, An Analysis Method with Failure Scenario Matrix for Specifying Unexpected Obstacles in Embedded System, The proceeding of the 12TH Asia-Pacific Software Engineering Conference, pp.447-454, 2005.
- [7] 渡辺博之、渡辺政彦、堀松和人、渡守武和記 “組込み UML”, 翔泳社 2002
- [8] 経済産業省 “2005 年版組込みソフトウェア産業実態調査報告書”