

組込みシステムにおける外部環境の分析

鷺見 毅[†] 平山 雅之[†] 鷗林 尚靖[‡]

[†]株式会社東芝 ソフトウェア技術センター

[‡]九州工業大学 情報工学部 知能情報工学科

組込みシステム開発には、厳しいリアルタイム制約、実世界へのリアクティブ性を持つといった、システムを構築する上で特有の難しさがあるが、リアルタイム性以外の特徴を対象とした分析手法は今までほとんど研究されていない。しかし、信頼性の高い組込みシステムを開発する為には、リアルタイム性以外の特徴に対しても、適切な分析を行う必要がある。

組込みシステムは、実世界の状態に応じて自身の状態や処理の内容を変化させながら動作する。その為、システムの分析段階において、システムの外部にある実世界についても分析しなければならない。その為には、組込みシステムが対応しなければならない実世界の状態(外部環境)を同定し、その上で影響を分析する必要がある。

本稿では、組込みシステムにおける“外部環境”を同定し、分析する方法を提案する。また、提案手法を例題に適用し、その有効性を確認した。

Analysis of the external environment for embedded systems

Takeshi SUMI[†], Masayuki HIRAYAMA[†], Naoyasu UBAYASHI[‡]

[†]TOSHIBA Corporation, Software Engineering Center

[‡]Department of Artificial Intelligence, Faculty of Computer Science and System Engineering,
Kyushu Institute of Technology

In an embedded system, there are several difficulties in its development - severe real-time restrictions, reactive restrictions from exterior environment and so on. However, research for analysis technique for embedded system mainly focused on only real-time restrictions of the embedded system. In order to develop a high-quality embedded system, the suitable analysis of the features other than real-time restrictions is required.

An embedded system is working, changing its state and process in concert with the state of real world. For this reason, we must identify the "external environment for embedded system" and analyze the effect, in system analysis phase.

In this paper, method to identify and analyze the external environment for embedded system is proposed.

1 はじめに

本稿では、ハードウェアを意識した組込みシステムの効率的な分析手法について述べる。

組込みシステムの最大の特徴は、ハードウェア

を用いて、実世界を対象とした機能を実現する点にある。この為、組込みシステムは実世界から様々な影響を受ける。これらの影響は以下の 2 つに代表される。

- ◆ 時間的な制約

数 msec～数 μ sec、或いはそれ以上に厳しいリアルタイム制約が掛かる。

- ◆ 実世界の物理事象に起因する制約

システムの外部にある空間(実世界)とデータをやり取りしながら、実世界の状態に応じて、システムの状態や処理を変化させる。即ち、実世界に対する応答性(リアクティブ性)を持つ。また、その動作は、システムが動作する限り持続する。

従って、実世界の状態や物理法則等によって影響を受ける。

このように、組込みシステムには、タイミングや実世界からの影響といった、システムの動作を変化させる要因が多く存在する。こうした要因に対応する為、組込みシステム開発における分析では、より高いシナリオの網羅性が求められる。その為、組込みシステム開発は分析時に考慮しなければならない事が多い。

これに対し現状は、リアルタイム制約にフォーカスした手法がほとんどで、実世界に対するリアクティブ性にフォーカスした手法はあまり提案されていない。しかし、実世界の様々な状態に対して適切に対応できる、より信頼性の高い組込みシステムを開発する為には、実世界に対するリアクティブ性を考慮し、システム外部にある実世界の影響を分析する必要がある。

本稿では、組込みシステムの実世界に対するリアクティブ性にフォーカスし、分析、設計上の現状での問題点について述べる。その上で、組込みシステムにおける実世界を同定し、組込みシステムに対する影響を分析する手法を提案する。

本稿の構成は以下の通りである。2章で関連研究について述べる。3章では組込みシステムと実世界との関連について、現状の問題点とこれを扱う上での特徴を述べる。4章で提案手法の概略について述べ、5章でその詳細について適用例を用いて説明する。6章で提案手法についての考察を述べ、7章で今後の課題を述べる。

2 関連研究

組込みシステムを対象とした開発手法では、eUML(組み込みUML)^[4]が有名である。eUMLは、

ハードウェアをクラスとする事により、ハードウェアとソフトウェアを同じ階層で扱う事を可能としている。しかし、実世界へのリアクティブ性についてはふれていない為、システムの動作が十分に網羅されていない可能性がある。

また、リアルタイム性については幾つかの研究がされている。リアルタイム構造化分析^[2]では、構造化分析・設計手法における、リアルタイム性について、その扱い方を述べている。また、最近では、UML2.0でリアルタイム制約を記述できるようにする研究がされている^[3]。しかし、いずれの研究も、リアルタイム性の議論が中心であり、実世界へのリアクティブ性については必ずしも十分には言及されていない。

3 問題点

3.1 組込みシステムと実世界の関連

組込みシステムは、システム外部にある実世界の状態をセンシングし、その結果によって動作を変えながらシステムの責務を果たしている。例えば、電気ポットは、ポット内の水温をセンシングし、その温度に応じて水を沸騰させる為にヒーターを作動させたり、水温が上がり過ぎないようにヒーターを停止させたりする。この場合、ポット内の“水温”は、電気ポットの動作にとって考慮しなければならない実世界の状態になる。

その為、実世界が取り得る可能性がある様々な状態に適切に対応できる、高い信頼性を持った組込みシステムを開発する為には、システムの分析を行う際に、実世界の状態も分析する必要がある。そして、システムの動作を記述するシナリオにおいて、より多くの実世界の状態を考慮する事、即ち、シナリオの網羅性を高める事が非常に重要となる。何故なら、電気ポットの例の様に、組込みシステムにおいては、システムが動作を決定する条件の中に、実世界の状態に起因する条件が含まれる場合があるからである。

以上の事から、組込みシステムの開発では、システムが実世界の状態をどう捉えているか、或いは、実世界の状態がシステムにどう影響しているかを分析する必要がある。しかし現状では、これらの分析手法は提案されておらず、十分な分析ができるかは分析者の力量に依存している。

そこで、本稿では組込みシステムに影響する実世界の状態の分析を、分析者の力量に依らずに行える手法の提案を目指す。

3.2 実世界の状態の特徴

本稿で扱う実世界の状態には、システムとは無関係に変化するという特徴がある。その為、予期しないタイミングで、予期しない状態になっている可能性がある。例えば、電気ポットで、ポット内の水を沸騰させる為にヒーターで加熱していても、ポットに冷水が追加された等の理由によって、(電気ポットにとって)唐突に水温が下がる場合もある。その様な場合であっても、組込みシステムは、変化した外部環境の状態に合わせて、自身の状態や動作を変化させ、システムが停止しないようにする必要がある。

また、その変化は、システムによってコントロールできないという事も特徴として挙げられる。システムの外部からの入力という点では、ユーザ操作が考えられるが、ユーザ操作がユーザインタフェース上で操作を禁止する等の処置により、ある程度はシステムの都合に合わせた入力がされるようにする事ができる。しかし、実世界の状態に対して変化を禁止する事は不可能である。従って、本稿で対象とする実世界の状態は、システムによるコントロールが難しいという前提で考えなければならない。

実世界の状態を対象とした分析手法を提案する場合には、以上の特徴を考慮する必要がある。

4 提案手法

4.1 提案手法における実世界の捉え方

我々は、組込みシステムで考慮しなければならない実世界の範囲を、「システムが持つ HW によってセンシングできるデータの範囲」とし、これをシステムの外部環境(以降、「外部環境」と記述)と定義する。そして、組込みシステムがセンシングしているデータを分析する事によって、外部環境の分析を行う手法を提案する。

高所から落とした際の衝撃等、センシングしていない範囲の影響も考えられるが、これらの影響の分析は、提案手法ではスコープ外とした。

4.2 用語定義

提案手法の説明の前に、提案手法で用いる言葉の定義を行う。

外部環境：

前節で述べた通り、組込みシステムがセンシングできるデータの集合と定義する。

HW 一覧表：

組込みシステムに登場する HW を記述する表(図 1)を指す。

なお、組込みシステムでハードウェアと言った場合には、CPU 等のプラットフォームとなる物と、電気ポットのヒーター等の周辺デバイスとなる物があるが、ここでのハードウェア(以降、単に“ハード”、又は“HW”と記述)は、後者の周辺デバイスとする。

表中には、HW 名、各 HW で扱われる論理データ(HW が扱うデータ)、HW が SW に対してデータの入力/出力側のどちらであるか(データ方向)、HW がデータ扱う上での制約事項(HW 制約)、システムが HW 制約を満たさない状態(逸脱状態)、逸脱状態の影響の大きさ(逸脱状態の影響度)が記入される。

HW名	HWが扱うデータ	データの方向	HW制約	逸脱状態	逸脱状態の影響度

図 1：HW 一覧表

HW 制約：

HW、またはシステムがデータを扱う上で守らなければならない制限、または、その動作上、特別な意味を持つ値を指す。

例えば、温度センサにおいて、センシング可能な“温度の範囲が-30℃から 60℃まで”となっていたならば、これは HW 制約になる。

逸脱状態：

HW 制約に対して、「システムがその制約を満たさない状態」を指す。

例外シナリオの発生条件に相当する。

4.3 提案手法の概略

提案する手法は、システム設計前の分析段階において、外部環境の影響を分析するものである。

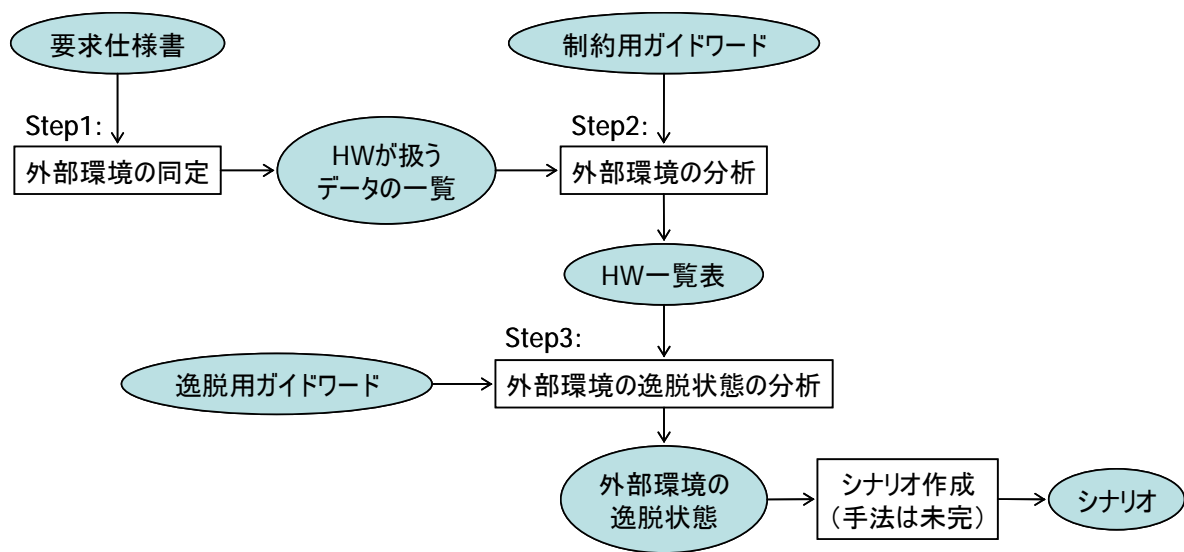


図 2：手法の概略

提案する手法の概略を図 2 に示す。図に示す通り、提案手法には、大きく以下の 3 つの Step がある。

Step1:外部環境の同定

システムに登場するハードウェアと、扱うデータを抽出する。また、データのソフトウェアに対する入出力方向を明確にする。

ハードウェアが扱うデータの集合を、外部環境とする。

Step2:外部環境の分析

Step1 で抽出した外部環境に対し、HW 制約を抽出する。

Step3:外部環境の逸脱状態の分析

Step2 で抽出した HW 制約に対して、制約を逸脱する状態が無いかを分析する。また、その影響の程度を分析する。

この手法により、以下の事を、分析者の力量に依存せずに行えるようにする事を目標とする。

- ◆ 外部環境の同定
- ◆ 外部環境の分析
- ◆ 外部環境によるシナリオの分岐条件の抽出

5 適用例

本章では、提案手法の詳細について、適用例を

用いて説明する。この適用実験には、SESSAME のセミナーで用いられている「話題沸騰ポット」の要求仕様書(第 3 版)⁵⁾を使用した。ここでは、「話題沸騰ポット」の沸騰・保温機能に提案手法を適用した結果を示す。

また、この適用実験の被験者は、組込みシステムの実開発を経験した事が無い技術者である。

話題沸騰ポットの要求仕様(一部のみを記載)

- ◆ 沸騰ボタンを押すと、ポット内の水を沸騰させてカルキ抜きを行う。沸騰中に押すと、沸騰を中止して保温状態になる。1 回押す毎に沸騰→保温→沸騰と変わる。
- ◆ 保温設定ボタンを押すと、保温モードを高温(98℃保温)、節約(90℃保温)、ミルク(60℃保温)モードに設定する。1 回押すごとに高温→節約→ミルク→高温とモードを変える。
- ◆ ヒーターの On/Off によって水温を制御する。
- ◆ ヒーターは、制御周期と操作量(%)によって制御される。
- ◆ 操作量 75%の時、制御周期の 75%の時間だけヒーターが On 状態になる。
- ◆ 沸騰状態では、ポット内の水を加熱し、100℃に達した後も 3 分間加熱を続け、その後保温状態に移行する。
- ◆ 沸騰状態が終了したら、水温を 98℃(節約モードでは 90℃、ミルクモードでは 60℃)に保

つようにヒーターを制御する。

- ◆ 水温が 110℃を超えた場合、ヒーター用電源を Off して 30 秒間ブザーを鳴らす。
- ◆ ヒーター制御中に 1 分毎に水温を検出し、目標温度よりも水温が 5℃下がり、かつ前回検出した水温よりも今回検出した水温が低い場合、ヒーター用電源を Off して 30 秒間ブザーを鳴らす。
- ◆ 満水センサ 1 つと水位センサ 4 つがあり、各センサが On の時、そのセンサ位置よりも水位が高い事になる。満水センサ On の時は、水位が許容上限を超えている事になる。
- ◆ 第 n 水位センサが On で、かつ満水センサが Off の場合、温度制御が可能になる。それ以外の場合には、沸騰ボタン・ヒーターは動作しない。

5.1 外部環境の同定

① HW 名の列挙

HW 一覧表の作成は、開発対象の組込みシステムの要求仕様書を参考に、組込みシステムで使用される HW 名を列挙する事から始める。コンシューマプロダクト(consumer product)の開発では、予め製品の仕様書の中で HW 構成を明示する事が多く、この作業は容易と考えている。

HW名	HWが扱うデータ
ヒーター	
サーミスタ	
第1水位センサ	
第2水位センサ	
第3水位センサ	
第4水位センサ	
満水センサ	
蓋センサ	
ヒーター用電源	
保温設定ボタン	
沸騰ボタン	

図 3 : HW 名の列挙

例題においても、要求仕様書に登場する HW が記述されており、そのまま HW 一覧表の HW 名の欄に記述した(図 3)。

② HW で扱うデータの明確化

次に、各 HW で扱うデータを明確にする。デー

タは、HW の仕様書から抽出する。また、HW が入力/出力しているデータの単位が何であるかを考える事で抽出する。HW 一覧表には、HW が扱うデータと、もしあるならばその単位を記述する。これにより、システムが実世界をどう捉えているか、即ち、組込みシステムの外部環境が明確になる。

ここでのデータは論理データであり、データ構造やその通信手段といった点は考慮しない。この分析結果はシステム設計の入力情報の一部として利用される為、論理データが明確になれば十分であると考ええる。

例題では、HW が扱うデータとして、以下のものを要求仕様書から抽出した(図 4)。

HW名	HWが扱うデータ	テ
ヒーター	On(操作量:%)/Off	
サーミスタ	水温(℃)	
第1水位センサ	On/Off	
第2水位センサ	On/Off	
第3水位センサ	On/Off	
第4水位センサ	On/Off	
満水センサ	On/Off	
蓋センサ	On/Off	
ヒーター用電源	On/Off	
保温設定ボタン	モード(高温/節約/ミルク)	
沸騰ボタン	モード(沸騰/保温)	

図 4 : HW が扱うデータの列挙

③ データの入出力方向の明確化

さらに、それらのデータがシステムに対して、入力として働くのか、出力として働くのかを明確にする。システムの入力/出力となるデータは、最終的にはシステムを制御するソフトウェアの入力/出力となる。通常、外部環境をセンシングしている HW が扱うデータは、システムを制御するソフトウェアにとって入力となり、ソフトウェアによって制御されている HW が扱うデータは、ソフトウェアにとって出力となる。従って、外部環境として捉えなければならないのは、ソフトウェアにとって入力となるデータである。

例題では、ヒーターとヒーター用電源の 2 つがソフトウェアからデータを受け取っている HW、それ以外は、ソフトウェアにデータを渡している HW になる(図 5)。

HW名	HWが扱うデータ	データの方向	H
ヒーター	On(操作量:%)/Off	Receive	
サーミスタ	水温(°C)	Send	
第1水位センサ	On/Off	Send	
第2水位センサ	On/Off	Send	
第3水位センサ	On/Off	Send	
第4水位センサ	On/Off	Send	
満水センサ	On/Off	Send	
蓋センサ	On/Off	Send	
ヒーター用電源	On/Off	Receive	
保温設定ボタン	モード(高温/節約/ミルク)	Send	
沸騰ボタン	モード(沸騰/保温)	Send	

図 5 : データの入出力方向の明確化

以上によって、ポットの外部環境は以下の様に特定できた。

- ◆ サーミスタにより検知される水温(単位:°C)
- ◆ 水位センサと満水センサの On/Off によって検知される水位
- ◆ 蓋センサの On/Off によって検知される蓋の開閉状態
- ◆ 保温設定ボタンと沸騰ボタンから送られるモードによって検知されるポットの状態

5.2 外部環境の分析

④ ガイドワードによる HW 制約の抽出

HW が扱うデータとその入出力方向が明確になった後に、各 HW がデータを扱う上で守らねばならない制約が無いかを、ガイドワードを用いて分析する。

ガイドワードを用いた分析手法は、化学プラントの設計の際に用いられる、HAZOP¹⁴⁾がある。HAZOPでは、化学プラントが災害を起こす可能性を網羅する為に、ガイドワードを用いた分析を行う。ガイドワードは、災害を起こす原因を端的に表しており(NO、MORE、LESS等)、災害に発展する状態(或いは、状況)が無いかを、プラントで扱う温度や圧力、制御フローにガイドワードを当てはめながら検証していく。

本提案手法では、このガイドワードを用いた分析を、HW 制約の抽出に応用する。

4.2 節で述べたように、本稿では HW 制約をシステムや HW がデータを扱う上での制限としている。そこで、“データを扱う上での制限”を端的に表す単語を“制約用ガイドワード”とする。例えば、センシング可能な温度の範囲が-30°Cから60°Cまでとなっていた場合、“から”と、“まで”がこれに相当する。

我々は、組込みシステムにおける HW 制約の抽出を目的とした、制約用ガイドワードとして以下のものを用意した。これらに該当する状態が無いかを検討する事により、HW 制約を抽出する。

- ~毎に (~単位で) per
- 上限、下限は~ upper/lower limit
- 連続して~の間 in a row
- 特別な値 special value

HW名	HWが扱うデータ	データの方向	HW制約	逸脱状態	逸脱状態の影響度
ヒーター	On(操作量:%)/Off	Receive	—	—	
サーミスタ	水温(°C)	Send	正常水温の上限=110°C	—	
			水温は1°C単位で計測	—	
			沸点=100°C	低気圧下では、沸点<100°C	
				不純物が混ざると、沸点>100°C	
			高温モードの保温温度=98°C	—	
	節約モードの保温温度=90°C	—			
	ミルクモードの保温温度=60°C	—			
第1水位センサ	On/Off	Send	Off=温度制御不可	—	
第2水位センサ	On/Off	Send	—	—	
第3水位センサ	On/Off	Send	—	—	
第4水位センサ	On/Off	Send	—	—	
満水センサ	On/Off	Send	On=温度制御不可	—	
蓋センサ	On/Off	Send	—	—	
ヒーター用電源	On/Off	Receive	—	—	
保温設定ボタン	モード(高温/節約/ミルク)	Send	—	—	
沸騰ボタン	モード(沸騰/保温)	Send	—	—	

図 6 : HW 制約と逸脱状態

制約用ガイドワードを用いた HW 制約の抽出は、HW 一覧表に記述されたデータの内、SW に対して入力されている(表中で Send となっている)データに着目する。

HW が SW に渡すデータについて、制約用ガイドワードで表されるような扱われ方をしないかを検討する事によって、外部環境を表すデータを扱う上での制約を得る。得られた条件は、HW 一覧表の“HW 制約”の欄に記入する。

これらの HW 制約は、システムが外部環境を扱う際に設けている“想定”と言える。逆に言えば、これらの想定を逸脱する状態は、システムに想定外の影響を及ぼす事になる。

例題では、図 6 に示す HW 制約の内、制約用ガイドワード“limit”によって“正常水温の上限=110℃”を、“per”によって“水温は 1℃単位で計測”を、残りの 6 個を“special value”によって抽出し、計 8 個の HW 制約を抽出できた。

5.3 外部環境の逸脱状態の分析

⑤ 外部環境の逸脱状態を抽出

抽出した HW 制約に対して、HW 制約を逸脱する状態が起り得ないかを検討する。そういった状態があるならば、逸脱状態として記録する。

逸脱状態は 4.2 節で述べた様に、「システムが HW 制約を満たさない状態」である。これを抽出する為に、さらにガイドワードを用いた分析を行う。ここでは、“HW 制約の逸脱の仕方を”端的に表した言葉を“逸脱状態用ガイドワード”とする。例えば、センシング可能な温度の上限が 60℃の場合、“60℃より高い”の“より高い”が相当する。

逸脱状態用ガイドワードとして、我々は以下のものを用意した。

- | | |
|-------------------------------|---------------|
| <input type="checkbox"/> ~が無い | no, not, none |
| <input type="checkbox"/> ~以上 | more |
| <input type="checkbox"/> ~未満 | less |

HW 一覧表に記述された HW 制約に対して、これらの逸脱状態用ガイドワードを当てはめ、該当する状態が起り得るかを検討する。この時、単体の HW 制約だけでなく、ユースケースに登場する複数の HW 制約を組み合わせる事によって、より複雑な逸脱状態を抽出する事ができる。

例題では、サーミスタの HW 制約である“沸点

=100℃”に対して、逸脱状態用ガイドワード more と less を当てはめる事で、“沸点が 100℃未満”と“沸点が 100℃以上”が抽出される。これに該当する状態が起り得るかを考えると、“低気圧下では、沸点は 100℃を下回る”、“ポット内の水に不純物が混ざっている場合、沸点は 100℃を超える”という条件で起り得る事が分かるので、これら 2 つが逸脱状態となる。これにより、以下の 2 つのシナリオが得られる。

- ◆ 低気圧(1 気圧未満)の状態では沸騰を開始すると、水温が 100℃に達しない為、いつまで経っても沸騰の終了条件が満たされない。
- ◆ 水に不純物が混ざっている状態で沸騰を開始すると、100℃で沸騰状態が終了する為に、水が沸騰しないまま保温状態に移行する。

この逸脱状態が抽出できないと、上記の条件に対してシステムが対応できなくなる可能性が出てくる。

⑥ 逸脱状態の影響度の分析

最後に、逸脱状態の影響度を分析する。逸脱状態の影響度は、その逸脱状態がシステム動作に対してどの程度の影響を及ぼすかを表すもので、以下の 2 つの観点からレベル付けを行う。

観点 1: 逸脱状態において、システムの動作(システム要求)はどうなるか

- 逸脱状態でも、ユースケースの Goal を達成できる
- 逸脱状態では、ユースケースの Goal を達成できない
- 逸脱状態では、ユースケースの Goal を達成できず、更に何らかのペナルティを被る

観点 2: 逸脱状態にならない為に、どの階層の対応が必要か

- Device Driver 等、比較的下位のレイヤーを変更する事で対応可能
- 逸脱状態にならない為には、シナリオを書き換える必要がある
- どうやっても、逸脱状態になってしまう

これらの観点を組み合わせると、以下の 7 段階のレベルができる。観点 1 にある“Goal が達成できる”場合は Level.0 とし、そのまま問題が無い事を表すとする。

- Level.0: ユースケースの Goal を達成できる
- Level.1: Goal を達成できないが、下位のレイヤーで対応が可能
- Level.2: Goal を達成できず、ペナルティを被るが、下位のレイヤーで対応が可能
- Level.3: Goal を達成できず、対応する為にはシナリオの書き換えが必要になる
- Level.4: Goal を達成できず、ペナルティを被り、対応には、シナリオの書き換えが必要になる
- Level.5: Goal を達成できず、対応も不可能
- Level.6: Goal を達成できず、ペナルティを被り、対応も不可能

レベルが高いほど、配慮が必要であるとする。得られたレベルは、HW 一覧表の逸脱状態の影響度の欄に記入する。

例題では、2 つの逸脱状態を共に Level.3 とした。これは、どちらの場合も正しく沸騰を終了できない事と、正しく沸騰を終了させるには、沸騰の条件を変えなければならない事から判断した。

6 考察

本稿では、組込みシステムにおける外部環境の影響を分析する手法を提案した。

外部環境の同定では、組込みシステムにおける外部環境を“HW によってセンシングできるデータの集合”と捉え、外部環境として分析しなければならない範囲を定義した。これにより外部環境として何を分析すれば良いかを明確にできた。その結果、外部環境の同定が可能になった。と考える。適用例では、サーミスタで検知される水温、センサの On/Off で検知される水位、センサの On/Off で検知される蓋の開閉状態、ボタンから送られるモードで検知されるポットの状態を外部環境として同定した。

また、外部環境の分析には、HW が扱うデータの中で、ソフトウェアに対する入力となっているデータに対して制約用ガイドワードを用いて分析する、HW 制約の抽出方法を提案した。適用例では、正常温度の上限が 110°C、沸点は 100°C、第 1 水位センサが Off の時は温度制御不可等、図 6 にある HW 制約を抽出した。

外部環境によるシナリオの分岐条件も、逸脱用

ガイドワードによる分析で抽出できた。適用例では、“低気圧下では、沸点は 100°C を下回る”、“ポット内の水に不純物が混ざっている場合、沸点は 100°C を超える”を逸脱状態として抽出した。

これは、組込みシステムが対応すべき実世界の状態を抽出できた事により、シナリオの網羅性を高める事ができたと言える。また、シナリオの分岐条件の候補である HW 制約と逸脱状態が明確になる事から、シナリオレビューの効率的に行えるようになったと考える。

また、適用実験の被験者が組込みシステム開発の経験が無い事から、分析者の力量に依存せずこれらの分析が可能になったと考える。

7 今後の課題

提案手法によって抽出されたシナリオの分岐条件を、実際のシナリオに反映させる手法については、今回はスコープ外としている。しかし、設計の入力としては、シナリオというかたちでシステムの動作が記述されている事が望ましい。抽出した条件のシナリオへの反映手法は、今後の課題としたい。

また、トライアルを行い、提案手法が従来手法を比較してより有効である事を示す必要がある。

参考文献

- [1] SAFEWARE, Nancy G. Leveson, Addison-Wesley Publishing Company, 1995
- [2] リアルタイム・システムの構造化分析, Derek J. Hatly, Imtiaz A. Pirbhai 著, 立田種宏 訳, 日経 BP 社, 1989
- [3] A UML Profile for Real-Time Constraint with the OCL, S. Flake, W. Mueller. UML2002-The Unified Modeling Language, Dresden, Germany, September/October 2002. Volume 2460 of Lecture Notes in Computer Science, pp. 179-195.
- [4] 組込み UML, 渡辺博之, 渡辺政彦, 堀松和人, 渡守武和記 著, 翔泳社, 2002
- [5] 組込みソフトウェア管理者・技術者育成研究会 (SESSAME) <http://www.sesame.jp/> (Last Access 2004/10/18)