平成 23 年度　戦略的基盤技術高度化支援事業（第３次補正予算）

「形式的仕様記述を用いた高信頼
ソフトウェア開発プロセスの研究とツール開発」

研究開発成果等報告書

平成 25 年 2 月

委託者　北海道経済産業局

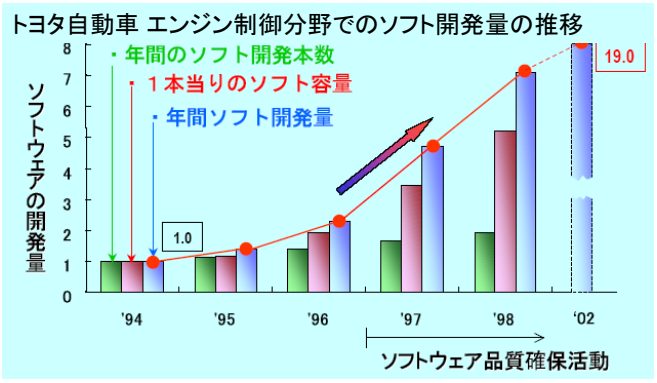
委託先　地方独立行政法人 北海道立総合研究機構

－ 目次 －

１． 研究開発の概要

１－１　　研究開発の背景・研究目的および目標

１－１－１ 背景



我が国が得意とする組込み機器製品の開発において、電子装置開発、特に組込みソフトウェア開発規模は爆発的に増大しおり、自動車ソフトウェアは８年で 19 倍（‘03/09 トヨタ自動車 重松;エンジン制御）に開発規模拡大が進んでいる。そのため、従来の開発技術によるソフトウェアの品質維持が困難となりつつあり、組込みソフトウェアに起因する重大故障が大きな社会問題となりつつある。

欧州では過去に発生した大規模事故（農薬工場の火災、石油タンク爆発など）から、電子装置に関する安全性および製品運用までを含めた製品ライフサイクルを厳密に定めた機能安全規格 IEC 61508 が策定され、欧州での電気・電子・プログラマブル装置による各種製品販売は機能安全規格への適合が必須となりつつある。この機能安全規格において、高い安全性を必要とする製品は、形式手法と呼ばれる技法を用いた開発が必須となっている。形式手法とは、1.数学的表現を利用した仕様記述により、曖昧性を排除する手法（形式的仕様記述），2. 制御の振舞いを数学的に表現し、その振舞いを網羅的に検証する手法（形式検証）である。

国内の電子装置の問題に目を向けると、近年、電子装置、特に、組込みソフトウェアに起因する重大故障が社会問題となりつつあり、組込みソフトウェアの不具合が重大な社会損失、企業経営損失を招いている（ 近年のソフトウェアに関する大規模リコールなど記憶に新しい）。

組込みソフトウェアが引き起こす不具合の原因として、1.要求仕様品質の問題、2.開発プロセスの問題、3.開発技術者のスキル問題、4.検証不良による問題などが挙げられる。上記問題のうち、開発プロセスは、品質および安全に関する規格にて規程を求めており、国内 IPA/SEC などでも有用なプロセス例を ExMR として紹介および推奨している。技術者スキル問題は、多くの企業および公的機関などでの教育等によりスキル向上の対策がとられている。また、検証不良に関する問題も、静的テストツール、動的テストツール、検証自動化ツールなど多方面からの対策が図られている。それに対して、要求および各種仕様の品質に関しては明確な対策が実施されていない。この問題解決に有効だと考えられる方法として、形式手法の一部である形式的仕様記述が注目されている。

形式記述に関する産業界の具体的ニーズとしては、自動車分野では、欧州のソフト標準化団体 AUTOSAR が ADL（アーキテクチャ記述言語）を基礎としてソフト部品化を推進している（鈴村・香月 : IPSJ SIGEMB 研究会’09 年 7 月）ほか、様々な欧州研究プロジェクト（EASIS、DEPLOY 等）で形式記述導入の検討が進んでいる。また、自動車分

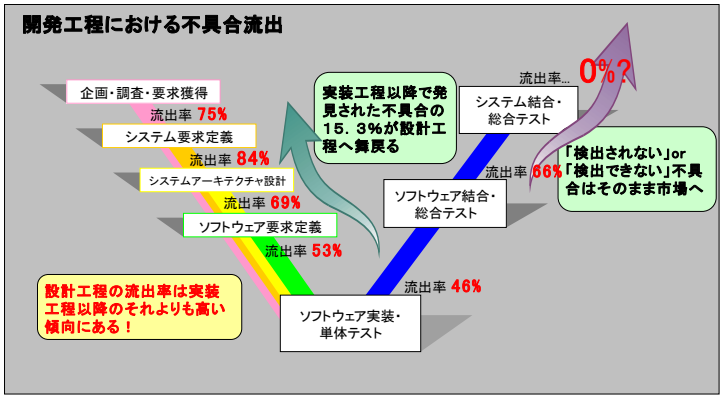
野の機能安全規格である ISO 26262 が正式制定され、形式記述の技術導入を含め、規格への対応が、欧州向け輸出を行う国内自動車関連メーカの急務となっている。

また、安全以外の分野としては、情報セキュリティ評価の国際規格 ISO 15408 においても特に高い水準(EAL6～7)では設計段階からの形式記述使用が推奨されており、運輸、電力などの重要インフラの関連システムや、将来的には個人情報等を扱う情報家電など分野でも対応が必要になると予想される。

1－1－2 目的と目標

本研究は、国内の組込みソフトウェア開発の現場における形式仕様記述の適用技術を確立させ、上流工程の品質向上、組込みソフトウェアの高信頼化を実現する事を目的とする。そのために、本研究では既存組込みシステムに利用されているアプリケーションプログラム、通信ミドルウェアを題材として、形式記述を実験的に導入し再開発する。また、その開発過程で得られた成果を分析し、技術導入を促進させるための形式的仕様記述支援ツールや教育プログラムを開発する。

現在のソフトウェア開発における問題点として、不具合混入されるプロセスの工程と不具合が発見される工程が異なり、ソフトウェア開発の手戻りが発生し、品質、納期、費用に大きな影響を与えている。すなわち、多くの不具合が開発上流工程で混入し、その検出が下流工程で検出されている。従って、品質確保は実装工程以降のテスト・検証工程が重要とされ、この工程で発見した不具合は上流工程に舞い戻る。また、不具合が混在したままで開発が進められるため、条件によっては検出されない・検出できない不具合も存在している。



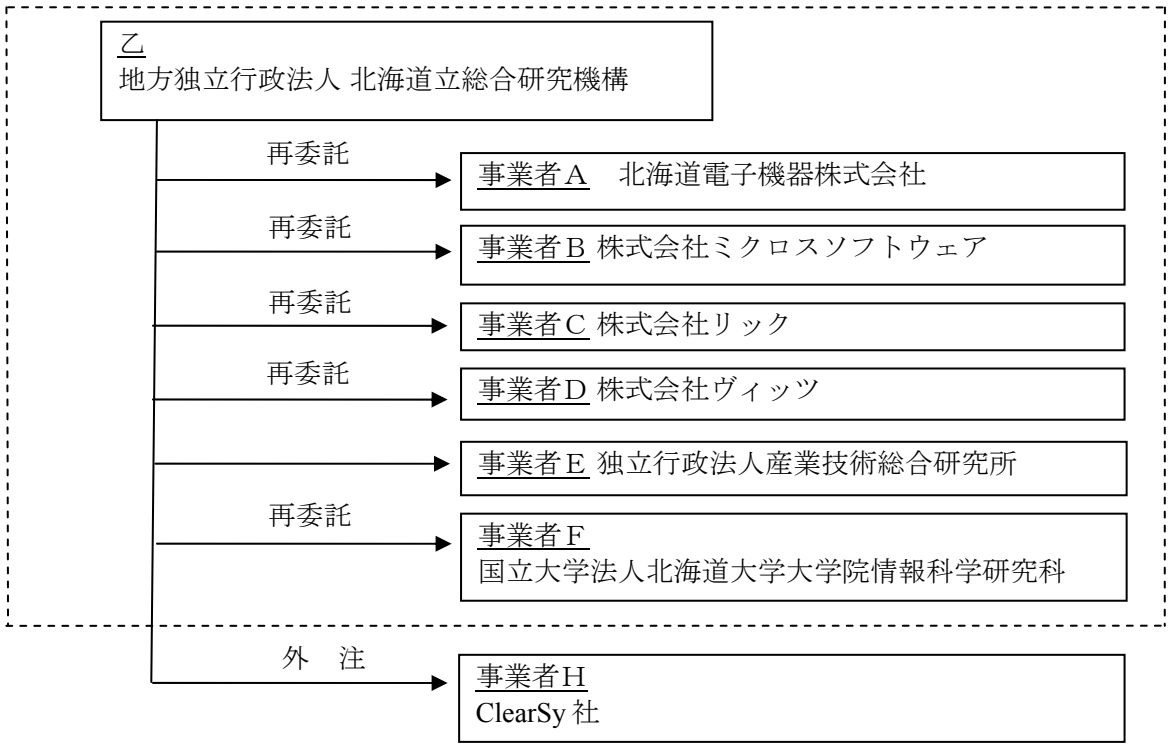
形式手法仕様記述の導入により、ソフトウェアの品質確保を上流工程で実現することで、下流工程への流出を減らし品質確保を早期に実現する。

ソフトウェアの品質確保を早期に行うことで、後工程での不具合発生も抑止でき、最終ソフトウェアの品質が格段に向上すると考えられる。

本研究は、この不具合流出率に着目し、現状のソフトウェア開発における平均的な不具合集流出率を大幅に改善し、全ての工程における流出率を半減させることを目標とする。

１－２　　研究体制

1)研究組織(全体)



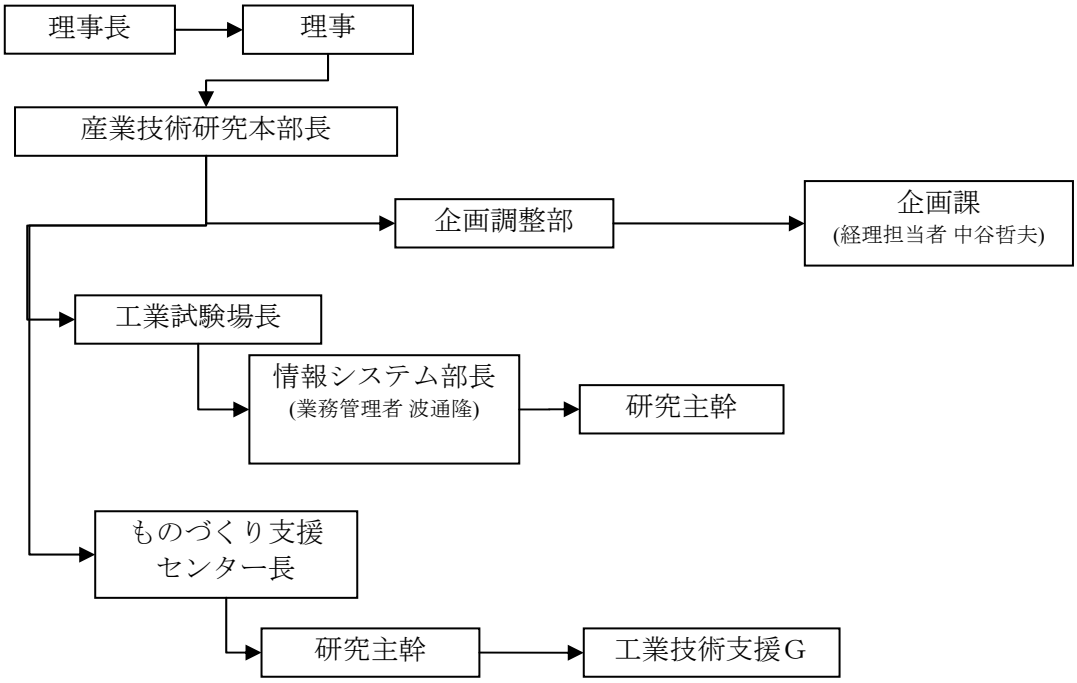
総括研究代表者（ＰＬ）
北海道電子機器(株)
技術部次長　穴田　秀樹

副総括研究代表者（ＳＬ）
(地独)北海道立総合研究機構
産業技術研究本部 工業試験場
情報システム部
計測・情報技術Ｇ
研究主任 堀　武司

2)管理体制

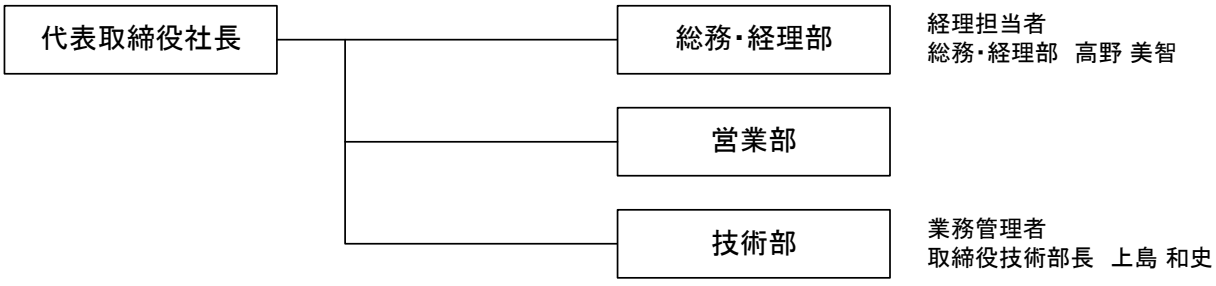
①事業管理者

地方独立行政法人 北海道立総合研究機構

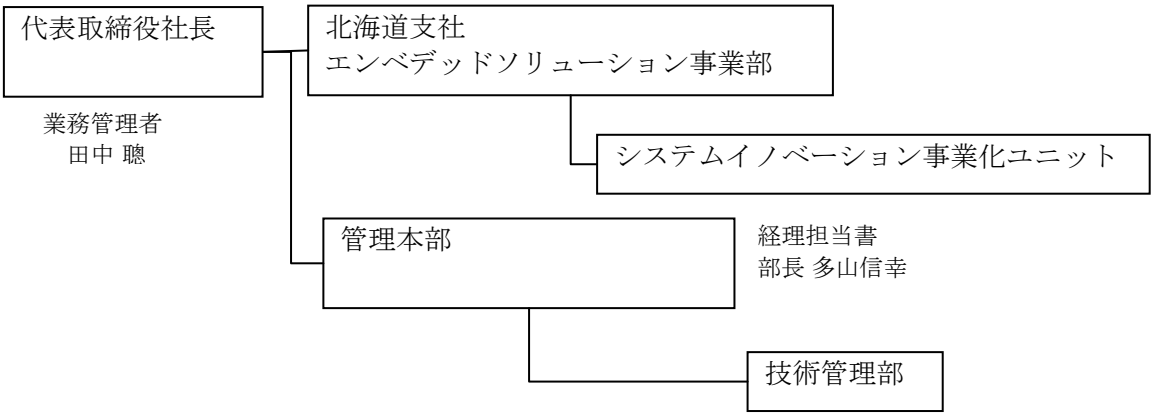


②(再委託先)

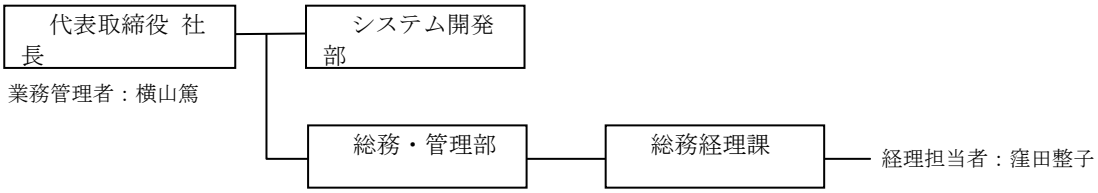
北海道電子機器株式会社



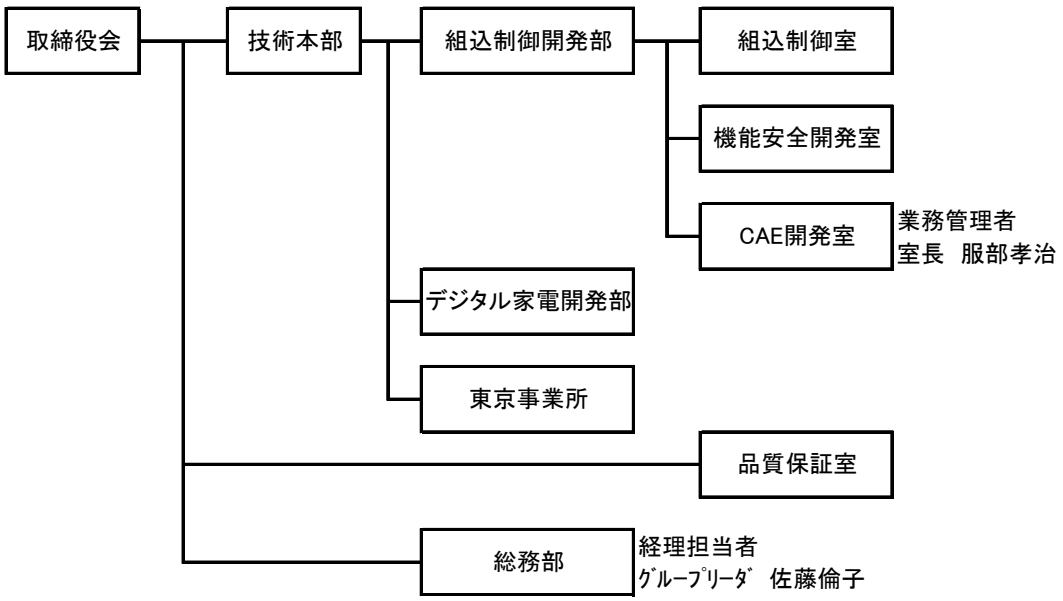
株式会社ミクロスソフトウェア



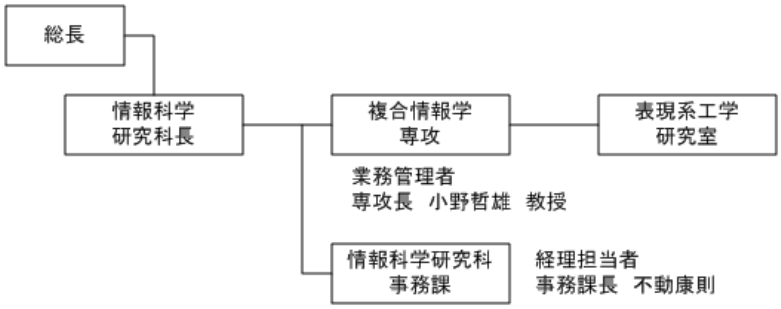
株式会社リック



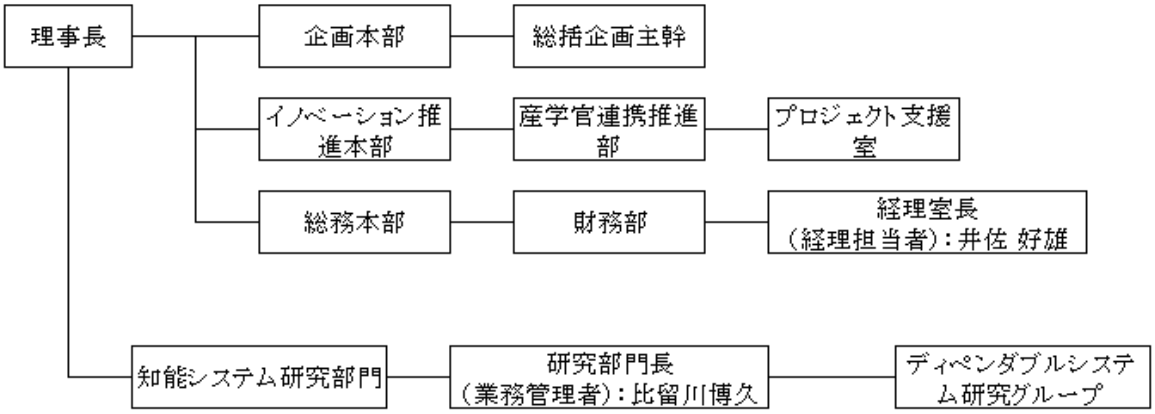
株式会社ヴィッツ



北海道大学



産業技術総合研究所



１－３　　成果概要

本研究の第一の目標は、国内の組込みソフトウェア開発現場における形式仕様記述の適用技術を確立させ、上流工程の品質向上と下流工程への不具合流出の削減（数値目標としては従来比半減）を実現することである。また、そこで得られたノウハウ等を利用し、国内での形式仕様記述を普及促進させることも、本研究のもう一つの目標である。

これらの目標を達成するため、本研究では以下の６つのサブテーマについて取り組みを行った。

サブテーマ１　形式仕様記述手法の調査

研究の初期段階において、B メソッド、Event-B 手法などの各種形式手法や関連する支援ツールの調査と、研究メンバへの内部教育を実施した。

また、サブテーマ２、３の試験導入開発の過程で発生した形式手法や B メソッドに関する技術課題について、関係者を集めた集中ワークの形式で問題解決を図り、これらの開発を側面より支援した。

なお、これらの過程で得られた情報は、サブテーマ６ 教材開発における基礎資料として活用された。

サブテーマ２　TCP/IP プロトコルスタックへの試験導入

組込み向け暗号通信ミドルウェアの開発を題材として、セキュリティ国際規格 ISO 15408 の規格要求事項に沿った形式手法の適用試験を行った。

ISO 15408 で求められる SPM(セキュリティ方針モデル化)、FSP(機能仕様書)、TDS(TOE 設計書)などの文書を自然言語と B モデルで記述し、さらに設計文書間の一貫性の証明を B メソッドの検証機能を用いて行う方法を提案した。また、作成した成果物文書等に関してフランス ClearSy 社によるレビューを受け、規格への適合方針や B メソッドのモデリングが妥当である事を確認した。これらの活動を通して、B メソッドによる ISO 15408 適合ソフトウェア開発の手法を確立することができた。

さらに、B モデルの一部について実装段階までの詳細化を行い、自動コード生成によるソフトウェア開発を試みた。その結果、実際に動作するソフトウェアを B で開発出来ること、不具合発生も抑止されている事が確認出来た。

サブテーマ３　自動車部品制御ソフトウェア開発への試験導入

アドバイザ企業から資料提供を受けた自動車部品制御の題材「ドアクローザ」に関して、B メソッドおよび Event-B の適用試験を行った。

(独)情報処理推進機構 から発行されている ESPR（組込みシステム向け開発プロセスガイド）をベースとして、それらの各工程に Event-B、B メソッドを用いた開発作業を導入した開発プロセスの修正案を作成し、それに沿って「ドアクローザ」ソフトウェアの設計・開発を行った。

V 字プロセスの左側(SWP1, 2, 3)では、Event-B や B メソッドを使ってモデル記述を行い、特に SWP2(アーキテクチャ設計工程)からコード生成までの作業を B メソッドによって一貫して実施可能であることが実証出来た。

テスト工程（SWP4、5、6）では、形式手法による品質改善効果を確認するため、従来開発と同様のテストを行った。その結果、テスト工程での不具合発見が 0 件となり、B メソッドの導入が下流工程への不具合流出の抑止に有効である事が確認出来た。

サブテーマ４　形式手法の導入効果分析

サブテーマ２、３で実施した試験導入開発の結果を分析し、形式的仕様記述導入による改善効果の評価を行った。

サブグループ３の例示開発では、グループ内および第三者による評価テストの結果では、いずれも残留不具合なしの結果が得られ、今回の事例においては不具合流出を 100%阻止することが出来た。また、サブテーマ２の例示開発においても、サブテーマ３ほど厳密な測定やテストは行っていないものの、同様に B メソッド適用箇所に関する不具合流出 0%との結果が得られた。

より詳細な評価として、ESPR の各工程で規定される検証プロセスの作業項目と、今回の事例で実施した作業を比較し、検証網羅率を検討した。その結果、SWP1～3 工程における仕様・設計の妥当性、一貫性、追跡可能性などに関して、B メソッドの検証機構によりほぼ網羅されている事が確認出来た。

サブテーマ５　形式記述支援ツール開発

B メソッドによる開発工程の一部を省力化するための支援ツールとして、自動生成されたソースコードを組込みマイコンボード上で動作させるためのコンパイル環境へ橋渡しするためのサポートツール「ビルド環境生成ツール」の試作を行った。国内の組込みソフトウェア開発における利用頻度の高いμITRON、OSEK/VDX の二つの組込み OS 環境、および OS なし環境に対応するツール試作を完了した。試作したツールは、サブテーマ６の教育コンテンツ開発やパイロットセミナで試用し評価を行い、その結果、B モジュール間の依存関係などを考慮して記述する必要があったモジュール初期化処理の呼び出しなど、煩雑な作業が自動化され、作業の効率化とケアレスミスの抑止に大きく貢献する事が確認できた。

サブテーマ６　形式記述教育コンテンツ開発

サブテーマ１～３の実施結果から得られた B メソッドのノウハウを整理し、技術者養成に活用するため、「基礎編」「応用編」「対話証明編」の 3 編の教材テキストを作成した。また、段階的詳細化などの B メソッドの利点を活用した実習課題として「電子施錠システム」の開発に関する例題を開発した。

これらの教材の評価を行うため、アドバイザ企業等の技術者を対象としたパイロットセミナを開催した。セミナの結果により、教材の改善すべき部分が明らかとなり、一部については教材の内容へフィードバックする事が出来た。

１－４　　当該プロジェクト連絡窓口

地方独立行政法人　北海道立総合研究機構
産業技術研究本部 工業試験場 情報システム部
計測･情報技術G
研究主任 堀　武司
Tel: 011-747-2942　email: hori-takeshi@hro.or.jp

２． 本論

２－１　　サブテーマ①: 形式的仕様記述手法の調査

２－１－１ 目的と目標

サブテーマ②③で実施する形式的仕様記述の試験導入において使用する形式仕様記述手法、および関連する文献、支援ツール、適用事例などの調査を行う。

２－１－２ 研究実施内容と成果

２－１－２－（１） 各形式手法の調査

研究開始の時点では、研究メンバの大半は形式仕様記述に関するスキルを持たない状態であったため、先行して取り組みを進めていた道総研などが中心となって、代表的な形式仕様記述手法の調査と、プロジェクト内での情報展開を行うとともに、試験導入で実際に試用する手法の選定を行った。

具体的には、VDM、B メソッド、および Event-B の 3 手法に関して、調査と内部報告会を行った。また、これらの手法を比較検討した結果、証明による厳密な検証が可能であり、かつ国内での取り組み事例がまだ少ない B メソッドを中心に進める事とした。

B メソッドは、VDM, Z 記法などと同じくモデル規範型形式手法と呼ばれるカテゴリに含まれ、一般のプログラム言語に比較的近い性質を持つが、その一方で定理証明に基づく強力な検証機能を持っている。また、段階的詳細化という考え方により、抽象的な仕様記述から、それと整合した設計・実装を段階的に導出する(Correct by Construction と呼ばれる)事が可能であるという特長を持つ。

その一方で、ヨーロッパを中心とする産業界での実績も豊富であり、パリ地下鉄の無人運転システムなどでの実績がよく知られている。また、モデリングや検証を行うためのツールチェーンも充実している。

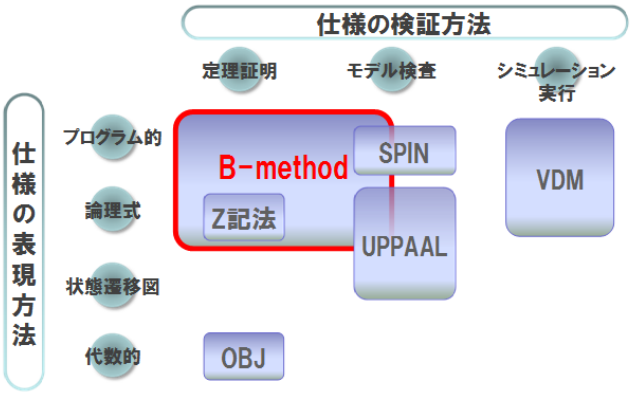


図１　代表的な形式手法の分類

２－１－２－（２） ワーク活動による技術課題の解決

２年度からは、サブテーマ②③におけるBメソッドを利用した開発活動が開始された。サブテーマ①では、それらの実施過程において発生した、形式手法およびBメソッドに関する技術課題を解決するため、関連する研究員による集中ワーク形式での検討を実施した。

主な集中ワークの検討内容とその結果を表 1に示す。。これらのワーク活動により、サブテーマ②③を進めるための技術課題を概ね解決することができた。

表 1　サブテーマ 1 で実施した主な集中ワーク

日時	ワーク課題	実施内容	実施結果	達成度
5/26	データリファインメント(G2)	通信プロトコル処理等で必要な複雑なデータ構造を、抽象記述→詳細化として記述する。	(当初のねらいとは異なったが) 定数に関する詳細化を実践出来た。	○
6/14	不変条件の記述方法(G3)	与えられた要求仕様から、有効な不変条件をどう抽出するか。	・(状態)=>(出力) の関係は不変条件として記述できるが、状態遷移規則などの動的な性質の記述は困難。	△
7/14	データリファインメント(G2)	通信セッション管理の仕様を例題として、データリファインメント記述を試みる。	・データリファインメントの方法は概ね理解出来た。 ・コード生成器の記述制約	○
8/1	時間仕様の記述 PO 爆発の抑止 (G3)	G3 開発における時間・タイマのモデリング方針を検討する。 PO 生成数の爆発的増加現象の原因究明と抑止方法の検討。	・時間に関しては、単純にカウンタを用いて記述する方式に決定した。（国内・海外調査でもそれで OK と確認) ・条件分岐の逐次処理的並びで PO 爆発する事を確認できたが、対処方法はワークでは確立出来なかった。	○ △
9/22	パケットデータの取扱い(G2)	・通信プロトコルにおけるパケットデータ等の複雑・大量のデータバッファの処理を、B でどのように記述するか？ ・B の厳しいスコープ規則をどう回避するか？	これらの課題を解決するため、データバッファ管理処理の一部を基本機械に委譲設計パターンを提案した。	◎
10/25	Event-B システムモデリング (G3)	G3 で試行錯誤が続いている SWP1 工程への B 導入方法の参考として、欧州 DEPLOY プロジェクトのシステムモデリング事例を調査検討する。	・仕様モデリングにおける段階的詳細化の使い方、結合度の低い Event-B モデリングの方法、モデルのデコンポジションの考え方等の知見が得られた。 ・しかし、現状の SWP1 モデルの構造と大きく異なる内容であるため、直ちに G3 開発に適用するのは困難と判断した。	△
11/11	FSP～TDS リフ	ISO 15408 対応開発にお	SPM(セキュリティポリシ)	◎

	ファインメント (G2)	いて、FSP(機能仕様)～TDS(設計)の内容の一貫性を B のリファインメント機構で表現する事を試みる。	～FSP～TDS に至るリファインメントを B で記述し証明するためのパターンを確立し、実際の G2 開発に反映できた。	
12/9	形式検証によるテスト代替 (G3)	G3 開発の各工程において B の形式検証機能を効果的に活用する方策を検討する。	・SWP3 工程において、ゼロ除算、オーバラン等といった不具合に関して、実際に PO が生成され証明で検証されている事を確認した。	○

２－２　　サブテーマ②: TCP/IPプロトコルスタック開発への試験導入

２－２－１ 目的と目標

本サブテーマでは、苫小牧高等工業専門学校で開発され、NPO 法人 TOPPERS プロジェクトから公開されている μITRON 用 TCP/IP 通信ミドルウェア TINET を対象として、形式仕様記述の試験導入を行う。

特に、通信で問題となるセキュリティ側面に関する形式記述の活用に着目し、コンピュータセキュリティのための国際規格 ISO 15408 を調査する。セキュリティレベルの評価指標である評価保証レベル（Evaluation Assurance Level, 略称 EAL）に基づき、形式手法の適用が要求されている箇所を特定し、その適用対象と適用方針について検討する。ISO/IEC 15408 で形式的な表現・証明が要求されている箇所において EAL6 の認証取得が可能なレベルで形式手法の適用を行う。（本研究では形式手法の試験導入を目的としている為、形式的な表現・証明に関連する要求のみを対象とする）

２－２－２ 研究実施内容と成果

２－２－２－（１） ISO 15408 規格の 調査

ISO 15408 の認証取得が可能な水準で形式手法を用いた開発の試行を行うため、ISO 15408 規格に関する調査を行った

ISO 15408 では、図 2に示すようにEAL1～7 の 7 段階の保証レベルで保証コンポーネントが定義されているが、それらのうち、形式手法の適用が求められているコンポーネントは赤枠で囲まれた EAL6、7 の保証コンポーネントに限られることが判った。

保証クラス	保証ファミリ	評価保証レベル別の保証コンポーネント						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
開発	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
ガイダンス文書	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
ライフサイクルサポート	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
セキュリティターゲット評価	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASE_REQ	1	2	2	2	2	2	2
	ASE_SPD		1	1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
テスト	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	3	3	4
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
脆弱性評定	AVA_VAN	1	2	2	3	4	5	5

図 2 レベル毎の保証コンポーネント一覧

表 2　形式手法に関連する保証コンポーネントの保証エレメント

保証コンポーネント	保証エレメント	解釈
ADV_FSP.6 追加の形式的仕様を伴う完全な準形式的機能仕様	**ADV_FSP.6.2D** 開発者は、TSF の機能仕様の形式的表現を提供しなければならない。	具体事例： ドキュメントに形式手法を適用した機能仕様を記載する必要がある。
	ADV_FSP.6.9C TSF の機能仕様の形式的表現は、適切な個所に対して非形式的で説明的なテキストで補足される形式的スタイルを使用して、TSFI を記述しなければならない。	形式手法を適用した表現の中に、適切にコメントを挿入する必要がある。
ADV_SPM.1 形式的 TOE セキュリティ方針モデル	**ADV_SPM.1.3D** 開発者は、モデルと任意の形式的な機能仕様との間の対応の形式的な証明を提供しなければならない。	形式手法を用いて、機能仕様がセキュリティを保つことを証明しなければならない。
	ADV_SPM.1.2C モデル化されるすべての方針について、モデルは TOE のセキュリティを定義し、TOE が安全ではない状態にならないことの形式的な証明を提供しなければならない。	形式手法を用いて、機能仕様がセキュリティを保つことを証明しなければならない。
ADV_TDS.6 形式的な上位レベルの設計提示を伴う完全な準形式的モジュール設計	**ADV_TDS.6.3D** 開発者は、TSF サブシステムの形式的仕様を提供しなければならない。	具体事例： ドキュメントに形式手法を適用した機能仕様を記載する必要がある。
	ADV_TDS.6.7C 設計は、目的、相互作用、インタフェース、インタフェースからの戻り値、及び他のモジュールに対して呼び出されるインタフェースの観点から、適切な箇所に対して、非形式的で説明的なテキストで補足される、準形式的なスタイルで各モジュールを記述しなければならない。	具体事例： ドキュメントに形式手法を適用した機能仕様だけでなく、順形式手法を適用した機能仕様を記載する必要がある。
	ADV_TDS.6.8C TSF サブシステムの形式的な仕様は、適切な個所に対して非形式的で説明的なテキストで補足される形式的スタイルを使用して、TSF を記述しなければならない。	形式手法を適用した表現の中に、適切にコメントを挿入する必要がある。
	ADV_TDS.6.10C TSF サブシステムの形式的仕様と機能仕様の形式的仕様間の対応の証明は、TOE 設計に記述されているすべてのふるまいがそれを呼び出している TSFI の正確かつ完全な詳細化であることを実証しなければならない。	機能仕様と形式手法を適用した TSF サブシステム仕様について、トサビリティを証明する必要がある。

本研究では、形式手法の適用が研究の主眼としており、それと直接関係の薄い規格要求事項全てを実施する事は、スケジュール、工数的にも困難である。そこで、本研究で取り扱う範囲を以下の様に設定した。

- 対象 EAL は、準形式手法、形式手法の使用が求められる EAL6 及び EAL 7 とする。

・ ST に関して、セキュリティ機能要件(Security Function Request、以下 SFR と呼称)及び、要約仕様が必要な保証エレメントに差し替える
・ 評価の対象を「開発（形式手法の適用が求められている箇所のみ）」「セキュリティターゲット評価」とする
・ 上記評価対象について、正式な認証ではなく第三者機関（評価機関）による模擬認証を受ける

次に、研究対象に含まれる保証コンポーネント間の関係や、それらに関連して求められる開発証拠書類について調査した。

図 3　「開発」クラスの保証コンポーネント間の関係

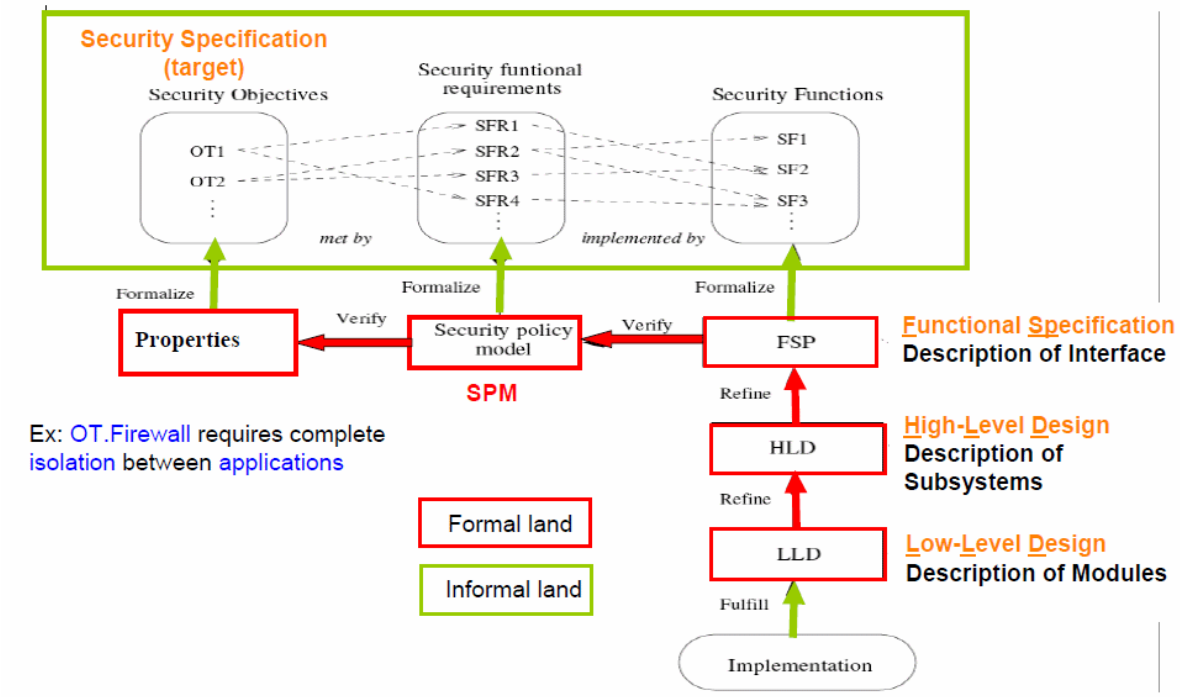


図 4　「開発」クラスの保証コンポーネントとその関係

調査の結果、全体の依存関係として図 4のような関係にあることがわかった。また、各保証コンポーネントと、セキュリティターゲットや他の保証コンポーネントの関係は以下のように定められていることが判った。

① セキュリティ方針モデル(ADV_SPM)

セキュリティターゲットでは、評価対象が対抗するセキュリティの課題をまず定義し、それに対して評価対象のソフトウェアがそれに対抗する為の対策方針とその為の運用環境などの前提条件を挙げる。そしてその対策方針がコモンクライテリアが定義するセキュリティ機能要件のカタログ中のどの機能に相当するのかのマッピングを示し、それがセキュリティ上の課題に対抗しうる理由を述べる。この部分の曖昧性を排除して厳密に表現をした上で、それがセキュリティ上の課題に対抗しうることを立証するのが SPM である。

セキュリティ対策方針はセキュリティプロパティと呼ばれることが多いようであるが、セキュリティ方針モデルの開発証拠書類ではセキュリティ機能要件の項目とセキュリティプロパティとの関係を定義して、セキュリティプロパティで定義した方針によりセキュリティ上の脅威に対抗できることを説明する。なお、外部機関によるレビューで確認したが、既存のセキュアなプロトコルを用いるような場合には、そのプロトコルそのものの安全性に関する証明はしなくてもよいとのことであった。

またセキュリティ方針モデルでは、機能仕様がセキュリティ方針モデルと一貫していることを証明する必要がある。セキュリティポリシーとの一貫性を機能仕様のコンポーネントではなくこのコンポーネント中で証明することが要求されているのは、セキュリティ方針モデルの形式化が必要な EAL レベルが機能仕様の形式化が求められる EAL レベルよりも下で有る為と考えられる。

② 機能仕様(ADV_FSP)

セキュリティターゲットでは、最後の章として TOE 要約仕様があったが、ここでは、評価対象のソフトウエアが提供するセキュリティに関わる機能を列挙し、それらのセキュリティ機能の中でセキュリティ機能要件(SFR)がどのようにして満たされるのかを説明をする。機能仕様では、このセキュリティ機能がどのインタフェースで、どのように使用されるべきものなのかを詳細に定義するものである。機能仕様はセキュリティターゲットの記載の厳密な詳細化となっている必要があり、またセキュリティ方針モデルとも一貫していることを証明する必要がある。

③ 設計(ADV_TDS)

設計は機能仕様を厳密に守った詳細設計へのブレークダウンであり、セキュリティ機能を提供するインタフェースがセキュリティ機能に関わる下位モジュールをどのように呼び出しているかの情報を提供する必要がある。サブシステムやモジュールの相互関係やサブシステム・モジュールのインタフェースの厳密な定義と共にこのセキュリティ機能のトレースを示すことが重要である。

２－２－２－（２） ISO 15408 適用開発

前節で行った ISO 15408 規格の調査、および規格の適用方針に従い、実際に B メソッドを用いて組込み通信ソフトウェアの設計・開発を行った。また、我々が作成した開発証拠書類や形式モデルが、ISO 15408 の形式手法関連の要求事項を満たしており、認証取得が可能な水準であるかどうかを判断するため、この分野に知見を持つ海外機関による開発成果物のレビューを行った。

本サブテーマにおける開発対象、およびISO 15408 における評価対象(Target Of Evaluation）として、TOPPERS OS と μITRON TCP/IP API 上で動作する「組込み用暗号通信ミドルウェア」（図 5）を選定した。ただし、暗号化ライブラリ、プロトコルスタック、およびカスタム組込みアプリケーションについては形式手法適用の対象外とし、HTTP over SSL/TLS のサブセット機能を実装する部分（図 5青色部）のみを開発の対象とした。

図 5　開発対象と開発範囲

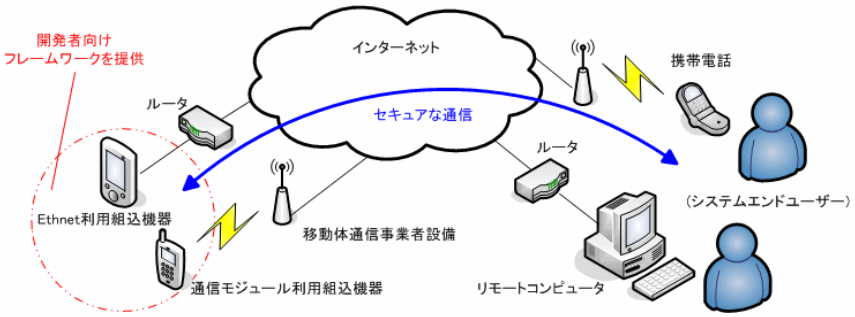


図 6 TOE の利用環境

この開発対象に関して、前節で決定したISO 15408 規格適用方針に従い、図 7に示す各設計文書の作成を行った。また、セキュリティターゲット以外の文書に関しては、自然言語による記述に加えて Bメソッドによる形式記述モデルもあわせて作成した。

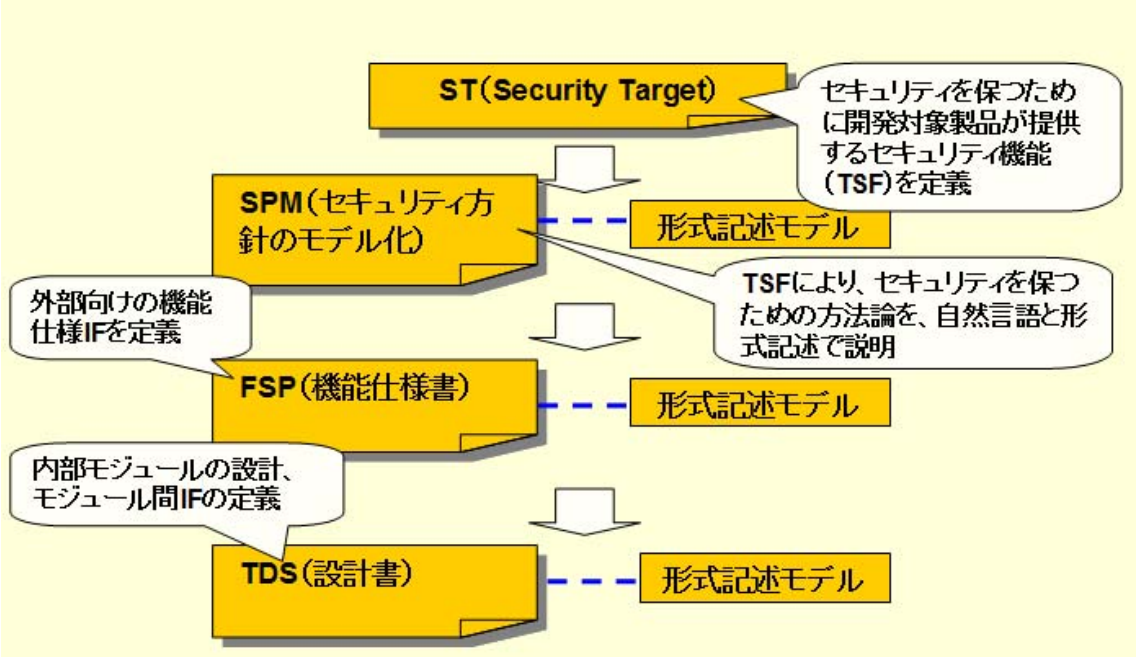


図 7 ISO 15504 に沿って作成した設計文書と形式記述モデル

ISO 15408 の規格要求では、これらのドキュメントの内容を単に形式的に記述するだけではなく、ドキュメント間の一貫性を「証明」する事が求められている。つまり、SPM でモデル化したセキュリティ方針が、FSP や TDS の記述と矛盾せずきちんと反映されている事の確認が求められている。

そこで我々は、図 8に示すように、Bメソッドの持つ段階的詳細化、およびその正当性検証の機能を用いることで、設計文書間の一貫性を示す方針を採用した。

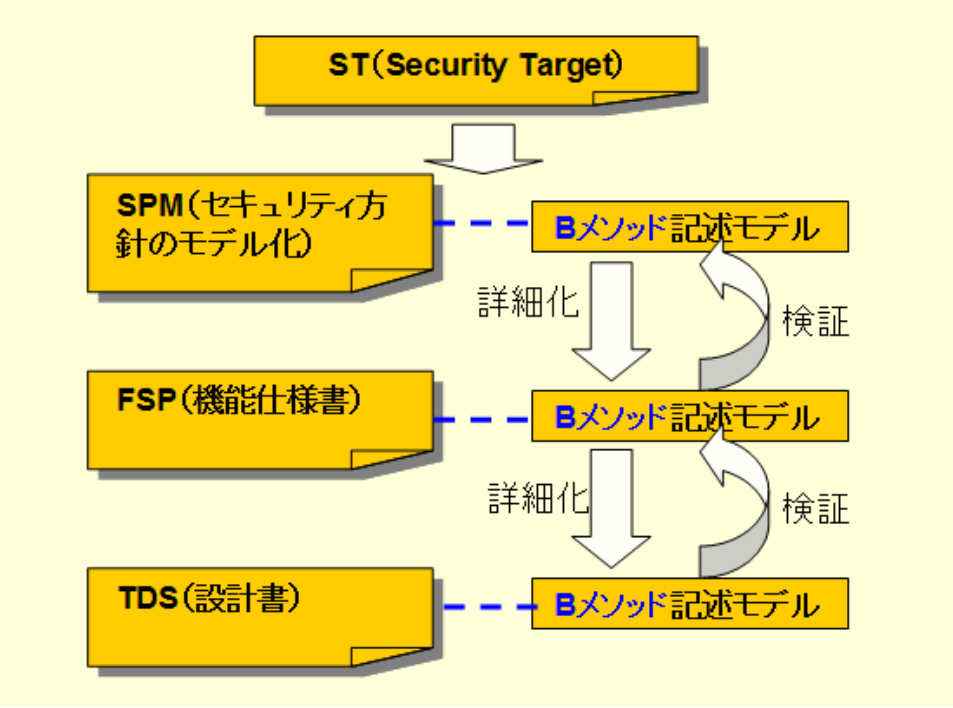


図 8 B メソッドの詳細化と検証機能を用いた一貫性の主張

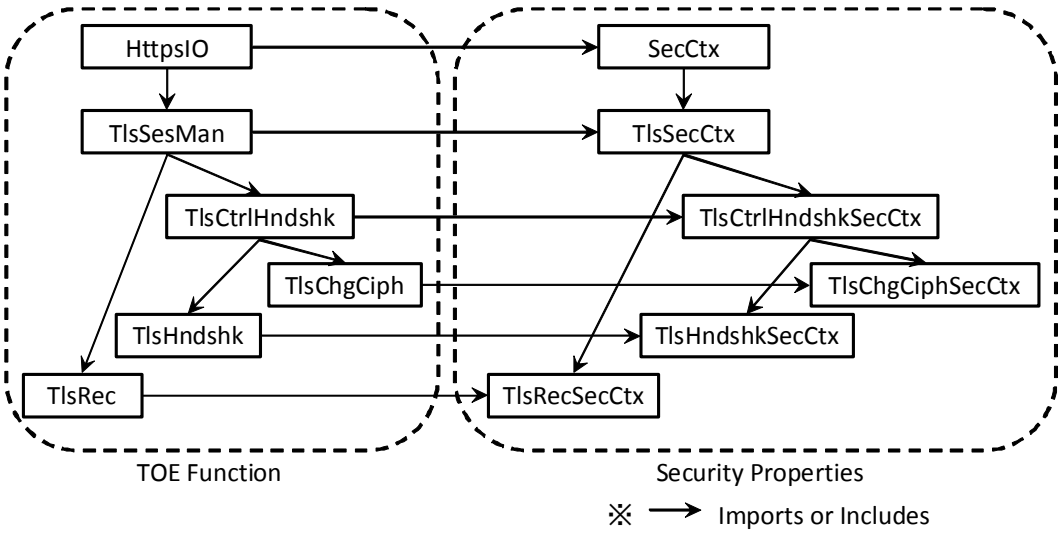


図 9 TDS(詳細設計)の B モデルの構造

図 9は、実際に作成したTDSのBモデルの構造の一部である。これらのモジュールのうち、SecCtx モジュールがSPM、HttpsIO モジュールがFSPの形式モデルに相当する。FSPであるHttpsIOはSPMであるSecCtxをインクルードしており、SecCtxで記述されたセキュリティ性質を満たす事が保証される。また、TDSモデルもHttpsIO モジュールの詳細化として記述されている。

２－２－２－（３） 海外機関による成果物レビュー

作成した設計文書、および B による形式モデルに関して、ISO 15408 規格（の形式手法関連部分）への適合性、および B メソッドとしてのモデル記述方針の妥当性について評価を行うため、フランス ClearSy 社による成果物レビューを実施した。ClearSy 社は、Atelier B ツールを開発するなど B メソッド実践の先進企業であると同時に、セキュリティや ISO 15408 規格対応に関する高い知見や業務実績を有しているため、本研究におけるレビュー依頼先として最適と判断した。なお、レビューは平成 23 年と 24 年にそれぞれ一回づつ、計二回実施し、研究の中間段階で妥当性を確認しながら進められるよう留意した。

図 10 に、最終レビュー報告書の一部を示す。ISO 15408 に含まれる形式手法関連の要求事項は全部で 24 項目（ADV_SPM、ADV_FSP、ADV_TDSの合計）あるが、最終的にはその全てに関して「適切」との評価を得られ、Bメソッドを用いた我々のISO 15408 規格適合方針は概ね妥当であることが確認出来た。

IV COMPLIANCY WITH COMMON CRITERIA REQUIREMENTS

The following table contains the result of the Common Criteria compliancy evaluation for the current project.

CC	Text	Explanation / conformance	Status
ADV_SPM .1.1C	The model shall be in a formal style, supported by explanatory text as required, and identify the security policies of the TSF that are modelled.	The syntax of B methods expressed the corresponding relationship between SFR and every security objectives. It also expressed how they confront to the threat. They are complemented with natural language in the section 2.1.	OK - checked.
ADV_SPM .1.2C	For all policies that are modelled, the model shall define security for the TOE and provide a formal proof that the TOE cannot reach a state that is not secure.	It identifies the requirements to keep security about all of security objectives of TOE in the section 2.1. It confirmed that the function of TOE fulfills all security properties in the section 2.4 by theorem providing.	OK - checked.
ADV_SPM .1.3C	The correspondence between the model and the functional specification shall be at the correct level of formality.	As it is described in the section 3.1, FSP formalizes fuctional specifications by using B method like SPM, includes modules which contains all of security properties defined in SPM. This means that security properties defined in SPM are all included in FSP.	OK - links between model and functionnal specification are easy to establish / understand.

図 10 ClearSy 社による最終レビュー報告書（一部）

また、上記 24 項目以外の ISO 15408 関連の指摘事項、および B メソッドのモデル記述方針に関する指摘事項についても、2 回のレビューの中でほぼ対応し、解決することが出来た。

２－２－２－（４） Bによるプログラム実装とテスト

B メソッドの特長の一つは、段階的詳細化とコード自動生成により、形式仕様記述から最終的なプログラム実装までを一貫して行う事が出来る事である。そこで、B で記述された TDS(詳細設計)モデルを更に詳細化し、実際に動作するプログラムを作成出来るかどうか、について試験を行った。

図 11にテストを実施した環境を示す。実験に利用可能な工数の関係により、HttpsIOモジュールのHTTPダイジェスト認証機能に限定して、Bによる（コンクリートモデルまでの）詳細化とComenCツールによる自動コード生成を行った。それ以外の部分については既存のソフトウェアを流用しB基本機械として実装した。また、サーバ側の動作環境は組込みマイコンではなく、VMware 仮想環境上のLinux を利用した。

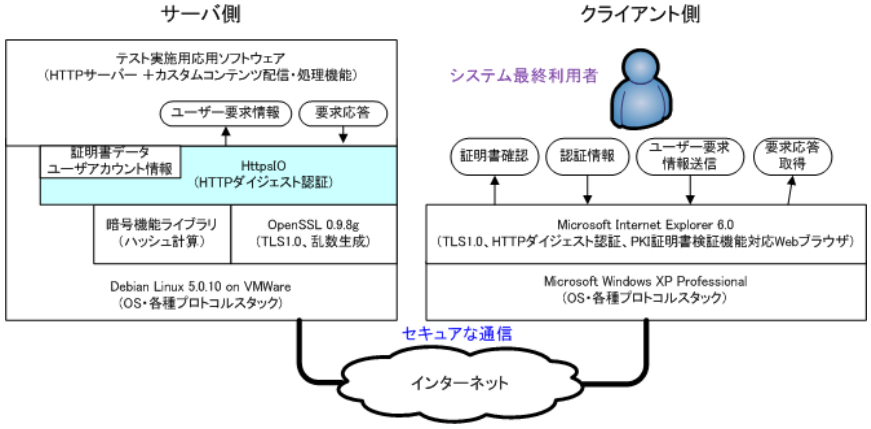


図 11 動作テストの環境

TOEを利用したテストアプリケーション（Webサーバ）に対して、外部Webブラウザから実際にアクセスを試みたところ、実際に動作し、かつ仕様通りの振る舞いをする事が確認出来た（図 12）。また、今回Bで記述したダイジェスト認証に関する項目についてソフトウェアテストを実施したところ、抽出したテストケース 43 件について不具合は 0 件であり、Bメソッドの適用によって実装段階での不具合発生が予防されている事が確認出来た。

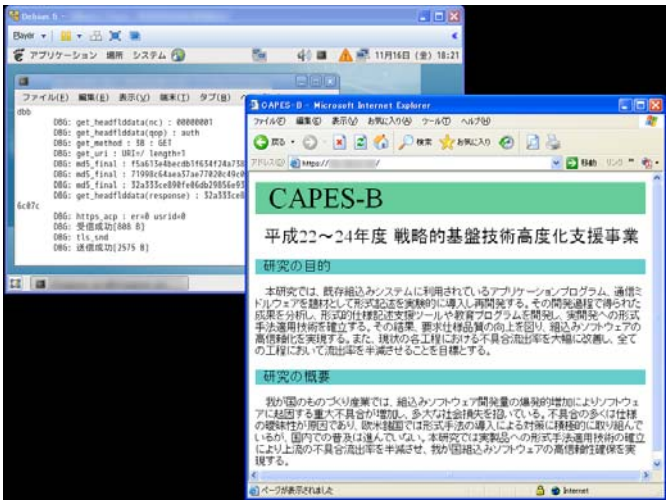


図 12 実装されたプログラムの動作試験

２－３　　サブテーマ③: 自動車部品制御ソフトウェア開発への試験導入

２－３－１　目的と目標

アドバイザ企業より要求仕様等の情報提供を受けた自動車部品の制御ソフトウェアを対象として、形式仕様記述を導入した開発プロセスを構築し、それに基づくソフトウェア開発の試行を行う。試験導入の過程において、組込み制御アプリケーションソフトウェア開発に形式的仕様記述を適用した場合の有効性を確認し、またその課題を明らかにする。同時に、自動車分野の開発に特有な課題を抽出し、それらに関する形式手法の効果的な活用方策も検討する。

２－３－２　研究実施内容と成果

２－３－２－（１）　形式手法を導入した組込み向け開発プロセスの構築

試験導入開発の実施に先立ち、B メソッド等の形式手法の活用を織り込んだソフトウェア開発プロセスの定義を行った。

ベースとなる開発プロセスとしては、(独) 情報処理推進機構 ソフトウェアエンジニアリングセンター(IPA/SEC)から刊行されている標準プロセス ESPR を用い、その各プロセスに形式手法導入に関す活動をマッピングする形でプロセス定義を行った(図 13)。

SWP1:ソフトウェア要求定義では、B メソッドの姉妹手法であり、より上流向けの Event-B を用いて要求仕様の検証を行う。また、SWP2:ソフトウェア・アーキテクチャ設計から SWP4:実装までの工程には B メソッドを適用し、段階的詳細化と定理証明により実装までの一貫性を保証する。実装作業は、B 言語から C へのトランスレータを使用する。

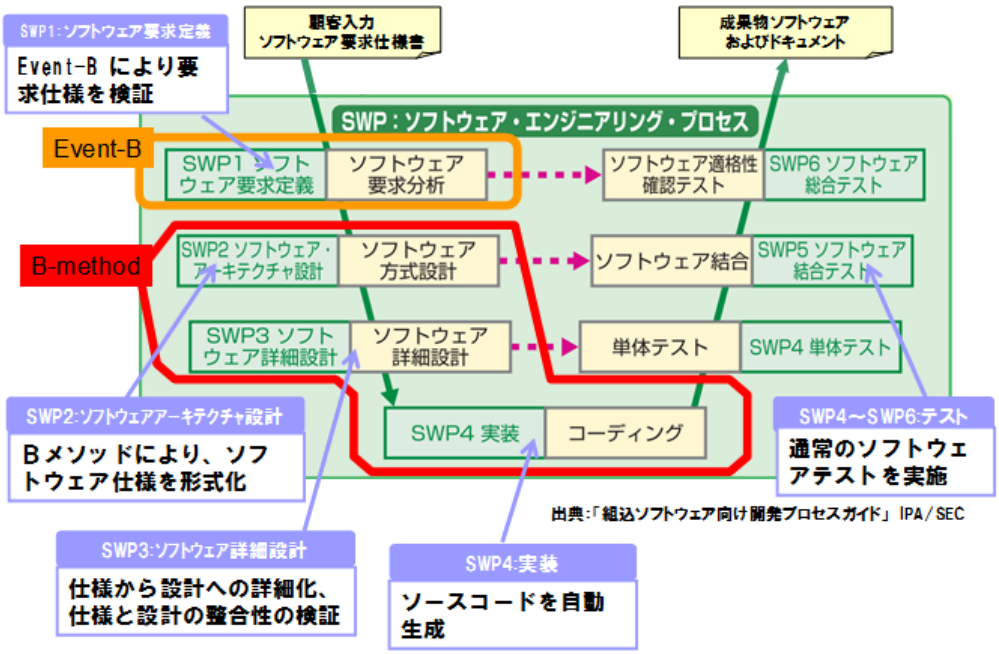


図 13　ESPR プロセスと B メソッド・Event-B の対応関係

２－３－２－（２）　ドアクローザの要求仕様定義

上で定義した開発プロセスに従い、本サブテーマの開発対象物である「ドアクローザ」の要求仕様の分析を行った。

まず、開発対象物の概略について説明する。「ドアクローザ」は典型的な小規模組込み制御システムであり、機構を動かす1個のモータと、メカの状態を検知する数個のスイッチ入力が、制御マイコンに接続されている。

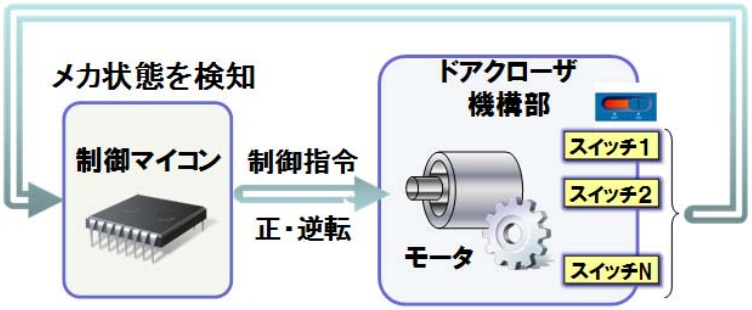


図 14　ドアクローザの機器構成

与えられたドキュメントの構成は、

・全体仕様書:（システム概要・関連ユニットのハードウェア仕様）

・制御仕様書:（状態遷移図、10～20 状態）

・参考資料:（関連ユニットのハードウェア仕様）

となっており、特に、制御仕様がかなり具象的な状態遷移図として与えられているのが特徴であった。

SWP1:要求仕様定義工程では、この状態遷移図を元に Event-B 手法と RODIN ツールを用いてモデリングを行い、動作シナリオに沿ったアニメーション実行などによって仕様の検証を行った。

２－３－２－（３）　ドアクローザの設計・実装

SWP2 以降の工程は、B メソッドと検証支援ツール Atelier-B で作業を行った。

SWP2: アーキテクチャ設計では、Bのモジュール分割によるアーキテクチャ記述と、仕様の形式記述への書き換えを行った(図 15)。この段階のモデルは、アブストラクトモデルと呼ばれ、この段階で仕様の形式記述への翻訳は完了となる。

SWP3: 詳細設計では、アブストラクトモデルのモジュール全てを完全に詳細化する作業を行った（図 16）。この状態のモデルをコンクリートモデルと呼ばれる。コンクリートモデルからは、Atelier-B に付属するトランスレータ ComenC により、C言語ソースコードを自動生成する事が出来る。そのため、SWP4:実装工程の作業は原則として存在しない。

本開発では、B コンクリートモデルから実際に ComenC によるコード生成を行い、得られたコードがドアクローザの実機ハードウェア上で動作する事を確認した。

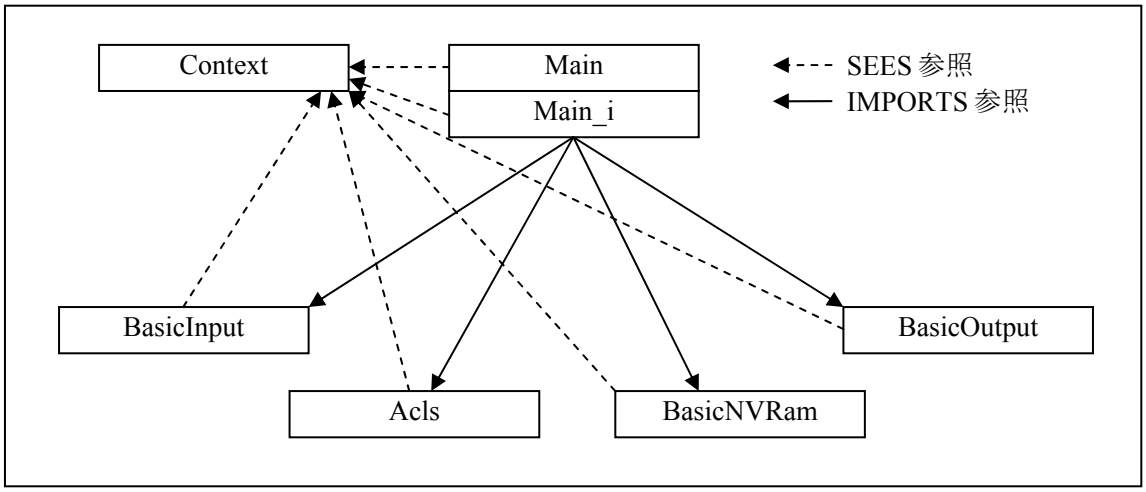


図 15　ドアクローザの B アブストラクトモデル

B 機械名	機能概要	備考
Context.mch	他の B 機械から参照する定数類(コンテキスト)を定義する。SW 類の ON/OFF 状態値定義、クローザモータ制御状態値定義などを含む。	抽象機械。
Main.mch	ECU ソフトウェア全体の 5ms 周期処理において、クローザ制御部の周期処理を担う。周期呼出インタフェースを提供する。	抽象機械。
Main_i.imp	本書に記載する動作シーケンス設計を実現するためのインプリメンテーション。アブストラクトモデルを構成する B 機械の各オペレーションを利用して、B 機械間の情報の受け渡しをおこなう。	インプリメンテーション。
Acls.mch	入力情報から出力情報を決定する状態遷移ロジックを担う。	抽象機械。
BasicInput.mch	既存の入力インタフェースを、B 言語記述に整合させる。	基本機械。
BasicOutput.mch	既存の出力インタフェースを、B 言語記述に整合させる。	基本機械。
BasicNvRam.mch	既存の保持 RAM インタフェースを、B 言語記述に整合させる。	基本機械。

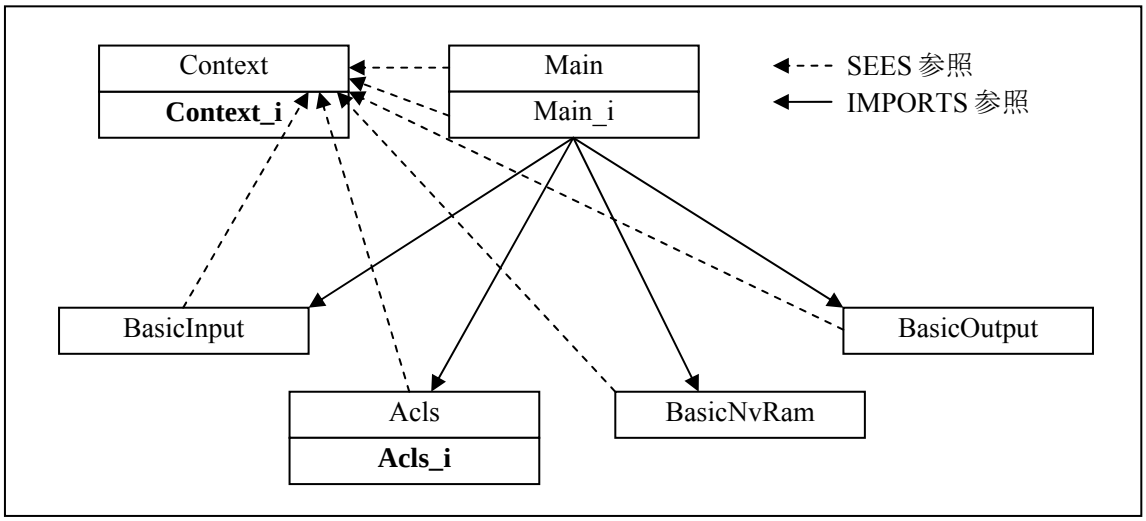


図 16　ドアクローザのコンクリートモデル

２－３－２－（４） ドアクローザのテスト

今回の試行の目的は、形式記述の導入による品質改善効果の評価である。そのため、SWP4～6 のテスト工程は従来型開発と同等の基準で実施し、その結果によって実際の品質改善効果を評価する事を試みた。

表 3～表 5は、単体、結合、総合テストの実施結果の集計である。いずれの工程においてもテスト工程における不具合発見件数は 0 件であり、(テスト漏れがないとすれば)実装工程より後方に流出した不具合は存在しないといえる。

これらのテストはグループ内で実施したものであるが、それとは別に、一部のテスト項目に関しては外部機関(自動車関連のソフトウェア企業)に委託して、実際の自動車ソフトウェア開発と同水準でのテストを実施した。その結果についても、内部での試験と同様に不具合発見は 0 件であった。

これらのテスト結果から、本開発では上流工程に B メソッドを導入した開発プロセスにより、実質的に不具合 0 件を達成できたものと判断した。

表 3　単体テスト実施結果

検査項目	抽出テスト項目数	NG 件数［件］	備　考
Main ユニット	**39**	**0**	**B メソッド**
BasicInput ユニット	**163**	**0**	**C 言語(基本機械)**
BasicNvRam ユニット	**8**	**0**	**C 言語(基本機械)**
Acls ユニット	**158**	**0**	**B メソッド**
BasicOuput ユニット	**12**	**0**	**C 言語(基本機械)**

表 4　結合テスト実施結果

検査項目	抽出テスト項目数	NG 件数［件］	備　考
インターフェース	**121/165**	**0/0**	センサ SW 入力処理
機能テスト	**41/49**	**0/0**	モータ制御
非機能テスト	**-/3**	**-/-**	スループット

表 5　総合テスト実施結果

検査項目	抽出テスト項目数	NG 件数［件］	備　考
クローザ制御機能	**200/169**	**0/0**	動作シナリオ

２－４　　サブテーマ④: 形式手法の導入効果分析

２－４－１ 目的と目標

本サブテーマでは、形式的仕様記述を試験導入して開発した各種ソフトウェアと既存ソフトウェアを比較し、形式的仕様記述導入による改善効果を定量的に測定する。それによって形式仕様記述導入によって得られる「ありがたさ」を明らかとし、形式仕様記述の普及促進を推進する事を目標とする。

具体的な活動については、以下の方針にそって実施するものとした。

１）導入効果を計測する対象は、サブテーマ①にて実施する各種ワーク活動、サブテーマ②、③で実施する開発とする。効果測定項目は IPA/SEC 発行の「IT の見える化」を参考に、開発工程毎に計測項目を規定する。重視する計測項目は工数・不具合数・不具合重大度・不具合トレーサビリティとする。

２）収集したデータを分析し、従来方法との比較や、形式的仕様記述の有効性をまとめ明らかにする。

２－４－２ 研究実施内容と成果

サブテーマ③の試験導入開発を対象として、開発プロセス関連データを収集して、導入効果の分析を試みた。サブグループ３でのテスト、および、第三者による評価テストを実施したところ、いずれも残留不具合なしという結果を得た。

また、サブテーマ②の試験導入開発に関しては、定量的な開発プロセスの測定データはないものの、B で実装まで行った部分のテストの結果では残留不具合なしとの結果が得られた。以上のことから、今回の二件の事例においては不具合流出を 100%阻止できたことになる。

ただし、この結果だけでは、Bメソッドの機能が具体的にどのように働いて不具合削減効果が得られたか、が明らかではない。そこで、さらに詳細な評価を行うために、ESPRにおいて規定されている以下の検証プロセスを対象として、今回の事例で実施した作業によってどの程度の検証網羅率を実現出来ているか、について、より詳細な分析を行った（表 6～表 8）。

- SWP 1.2 ソフトウェア要求仕様書の確認
- SWP 2.2 ソフトウェア・アーキテクチャ設計の確認
- SWP 3.2 ソフトウェア詳細設計の確認

表 6　ソフトウェア要求仕様書の確認における検証網羅率

内容	形式手法による実施作業	形式手法による検証網羅率	備考
①妥当性を評価する	・イベントＢモデルを用いたアニメーションにより、状態遷移図とメカ機構の間に矛盾がないことを確認 ・SWP6 の妥当性確認を前倒しできた（6 から 1 への手戻り防止）	・シナリオ（26 件）受理率は(26/26)100% ・総合テストの前倒し率 100%	・シナリオの網羅性が十分であることを仮定 ・要求仕様書が完全にソースコードに引き継がれていることを仮定
②実現可能性を評価する	（未実施）		
③テスト可能性（ソフトウェア要求はテスト可能か）を評価する	機能要求のテスト設計については、①が完了するなら、③は満たされる。		
④運用・保守性を評価する	（未実施）		
⑤追跡可能性を評価する	イベント B モデルの構成要素にトレーサビリティ ID を付与している	・モデルへのトレーサビリティは 100%	・受理したドキュメントからの要件の抜出しが十分であることを仮定
⑥一貫性を評価する	PO の証明により、要求仕様書の内部矛盾がないことを確認した（インバリアント、型）。	・PO の証明率は 100% ・インバリアントに由来する PO の証明率は 100% ・演算の well-definedness は、100%	・B メソッドが仮定する一貫性は証明可能 ・インバリアントが十分であることを仮定
⑦完全性を評価する	・イベントＢモデル記述とアニメーションを通じて、遷移条件の非決定性を 2 件指摘した（イベントＢモデルの記述ミス） ・イベントＢモデルの記述は、解釈が一意に定まる	・意図しない非決定性排除率 100% ・解釈の一意性は、100%	

表 7　ソフトウェア・アーキテクチャ設計の確認における検証網羅率

内容	形式手法による実施内容	形式手法による検証網羅率	備考
①ソフトウェア・アーキテクチャ設計書（SW205）の内容が適切であるかどうかを確認する。	▼機能ユニットの妥当性について： ・SWP1 の要件一覧と、SWP2 アーキテクチャ設計（アブストラクトモデル）との間のトレーサビリティは取った。 ・12 状態とモーター出力の関係をインバリアントとして記述して、証明した（スタティックな部分）。 ▼機能ユニット集合の振舞いの妥当性について： ・アブストラクトツリーの記述および PO 証明により、要件からのトレーサビリティおよび一貫性のあるモデルを得た。 ▼結合テストの前倒しについて、SWP5 でのテストケースと、B モデルとの対応を確認した。	・機能要件トレーサビリティは、100% ・スタティックな部分は、検証率 100% ・ダイナミックな部分は、実績としてはあまりできていない。しかし海外調査を含めて得られた知見から判断すると、改善可能 ・PO は 100%自動証明（証明工数 0h）	・並行処理動作（マルチタスク）の検証については、B の範囲外と考える。 ・アーキ記述が完全で、Correctness by Construction が正しく行われるなら、テスト結果は保証される。
②機能ユニットの詳細化設計を確認する	▼機能、インタフェース、振る舞いの妥当性： アブストラクトモデルの記述および整合性証明により確認した。 ▼ 具体化レベルと詳細設計の実現可能性： 外部仕様レベルで矛盾がある場合は、B モデルの記述・証明により摘出できる。 アブストラクトモデルにより、何を実現すべきかを、明確・局所的に提示でき、実現可能性の判断が容易になる。 ▼ 設計標準への準拠（設計手法/表記法/用語/分かりやすさ/名称の体系化）： B メソッドの規則に従うことが強制される。変数等の命		・モデリングガイドラインを作成中

	名規則を作成した。アーキテクチャ標準を確立した。		
③ソフトウェア要求との対応（トレーサビリティ）	（未実施）		

表 8　ソフトウェア詳細設計の確認における検証網羅率

内容	形式手法による実施内容	形式手法による検証網羅率	備考
①下記の視点でソフトウェア詳細設計書の内容を確認する。	▼機能仕様を正しく詳細化していることを、コンクリートモデルの記述および、PO の証明により、確認した。 ▼機能ユニット間インターフェースは、SWP2 で整合性を確認したものを、引き継いでいる。 プログラムユニットインターフェースについては、今回は考慮不要 ▼ハードウェアとのインターフェースは、基本機械の操作として集約した。これにより、インターフェース仕様を明確にした。かつ、インタフェースを外部オペレーションから正しく呼び出すことは、証明済み。	▼機能仕様に対する検証率 100%（トレーサビリティの確実度が 100%）	▼実装の効率（時間、メモリ）については、レビューが必要かもしれない。
②内部確認レポート（SW306）	（未実施）		

２－５　　サブテーマ⑤: 形式記述支援ツール開発

２－５－１ 目的と目標

サブテーマ⑤では、サブテーマ②、③における試験導入開発で得られた開発上の課題点のうち、支援ツールによる作業の自動化、省力化により解決可能な課題を抽出する。さらに、分析結果に基づき支援ツールの仕様検討、および核となる部分の試作を行い、事業終了後の製品版ツールの開発の準備を行う。

具体的な支援ツールの内容としては、他手法に対するBメソッドの優位点である実装段階での自動コード生成機能に着目し、国内の組込み制御分野において普及度が高いOS(オペレーティングシステム)であるμITRON、および自動車制御分野で国際標準になりつつあるOSEK/VDX OSの二つの環境を対象とした、Bメソッド自動生成コードの実装作業支援ツールとする。

２－５－２ 研究実施内容と成果

２－５－２－（１） C4Bトランスレータの調査

Bメソッドの統合開発環境「AtelierB」には、B言語をC言語に変換する機能が備わっており、これを「トランスレータ」と呼ぶ。ビルド環境生成ツールは、C言語ソースコードをビルドするための環境構築を補助するものである。したがって、トランスレータが具体的にどのようなC言語ソースコードを生成するか把握しなければならない。

AtelierBの現バージョン(version 4.0)ではトランスレータとしてComenCツールが付属しているが、トランスレーション可能な記述に制約が多く使いづらい面があるため、今回の開発ではAtelierB次期バージョン(version 4.1)で用いられる次世代トランスレータ「C4B」の使用を前提にツール開発を行う事とし、そのために必要なC4Bツールのコード生成規則等の調査を実施した。

調査結果の一部を表9に示す。このように、B言語の要素ひとつひとつに関してC言語への変換規則を確認し、支援ツールや基本機械ライブラリを作成するために必要な情報を全て取得する事が出来た。

表9　C4Bにおける B言語から C言語への変換規則（一部）

No	B言語 Operation	意味	C言語変換規則
1	.	名前の付け替え	C4Bで変換不可
2	$0	代入前の変数値	C4Bで変換不可
3	()	括弧	()は()に変換される
4	""	文字列	“abc”は”abc”に変換される
5	TRUE	真値	TRUEは1に変換される
6	FALSE	偽値	FALSEは0に変換される
7	bool	述語から式への変換演算子	C4Bで変換不可
8	MAXINT	実装可能な整数の最大値	MAXINTは2147483647に変換される
9	MININT	実装可能な整数の最小値	MININTは-2147483647に変換される
10	add	加法演算子	addは+に変換される
11	sub	減法演算子、負号	subは-に変換される

12	mul	乗法演算子	mul は*に変換される
13	div	除法演算子	div は/に変換される
14	modu	剰余	modu は%に変換される
15	x**y	べき乗（x の y 乗）	x**y はべき乗関数に変換される
16	succ(x+y)	後継者	succ(x+y)は(x+y)+1 に変換される
17	pred(x+y)	前任者	pred(x+y)は(x+y)-1 に変換される
23	|->	マップレット	|->を使って定義したマップレットは配列データとして変換される。 例） var5 := { 0 |-> -4, 1 |-> -5 } int Expressions__var5[2] ={- 4, - 5};

２－５－２－（２） ビルド環境生成ツールの開発

B メソッドは、ソフトウェアアーキテクチャ設計工程（SWP2）から実装工程（SWP4）までを、統合開発環境「AtelierB」を利用して一貫性を保ったまま開発できるという長所がある。その一方で、実装工程の成果物である C ソースコードをビルドする環境構築は、AtelierB のサポート範囲外であるため手作業で実施しなければならない。ビルド環境構築は、ターゲット OS に関する知識を必要とし、また手作業は人為的ミスを誘発する危険がある。そこで、B メソッドで開発した C ソースコードを、代表的な組込み OS である μITRON および OSEK/VDX 環境向けにビルドするための環境構築を補助する「ビルド環境生成ツール」を開発した。

開発した「ビルド環境作成ツール」の構成を図 17に示す。

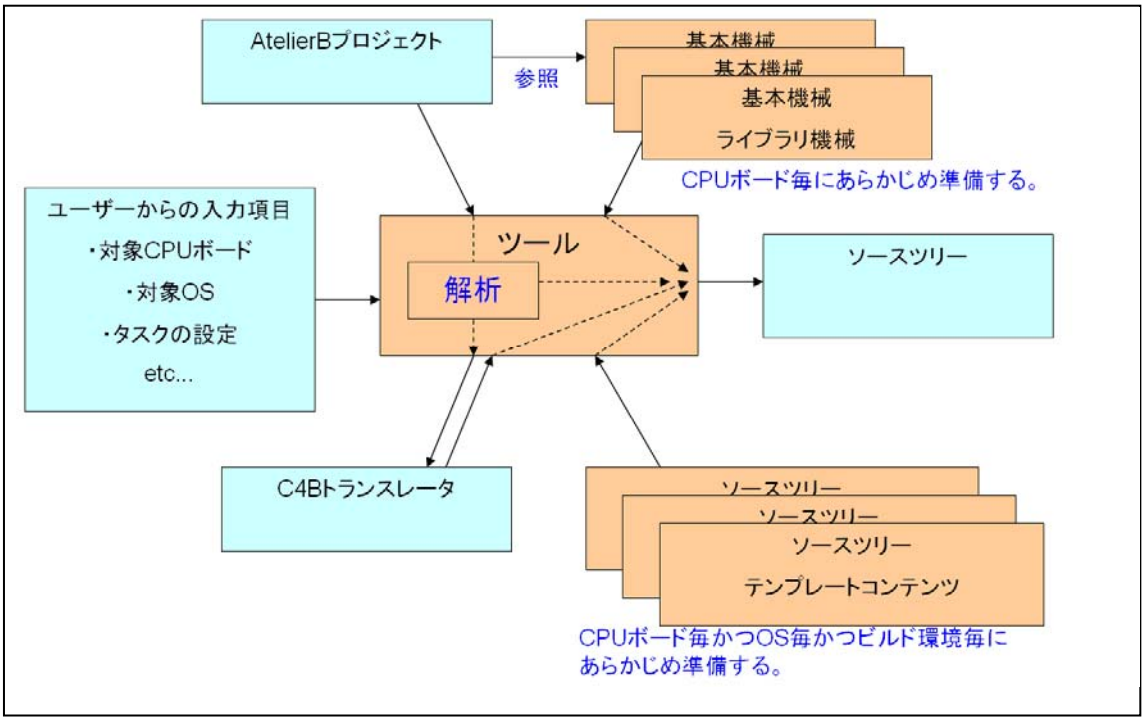


図 17　ビルド環境生成ツールの構成

環境生成ツールは、様々な CPU ボード、様々な OS 向けのソースツリーを生成できることが望ましい。これを実現するために、CPU ボードごと、OS ごとに、ソースツリーのテ

ンプレートを追加できる設計（テンプレートコンテンツ）とし、この機能を用いてμITRON、OSEK/VDX、およびOS無し環境向けのソースツリー生成機能を実装した。μITRONおよびOSEK/VDXの具体的実装としては、オープンソース製品であるTOPPERS/ASPおよびTOPPERS/ATKを対象とした。また、CPUボード依存のデバイスドライバに関しても基本機械のライブラリとしてパッケージ化し、対象とするCPUボードに合わせて複数から選択可能な設計とした。

Bメソッドで開発したソフトウェアモジュール（以降「Bモジュール」）をその外部から駆動する部分は、通常はC言語などにより手作業で記述しなければならないが、本ツールでは駆動処理部のソースコードを自動生成してソースツリーに含める設計とした。駆動処理部では、定常周期実行処理に移行する前に、各Bモジュールの初期化関数を呼び出す必要がある。初期化関数の呼び出し順序は、各Bモジュール間の関係に依存して一定の規則に従わなければならない。それに必要な各Bモジュール間の依存関係の解析を図17中の「解析」部で実施する設計とした。

「ビルド環境作成ツール」を利用した開発手順は、概ね次のようになる。。

① ユーザーはAtelierBを利用して、実装工程（SWP4）まで実施する。
② その後ビルド環境生成ツールを起動し、対象とするOS、CPU、および開発環境などを指定し、ソースツリーを生成する。
③ 対象マイコン用の開発ツールを用いてソースツリーのビルドとロードモジュール作成を行う。

以下、ツールの実行画面を示しながら詳細を説明する。

本ツールはAtelier Bのプラグインとして実装されており、AtelierB上のプロジェクトメニューから直接起動する事が出来る（図18）。

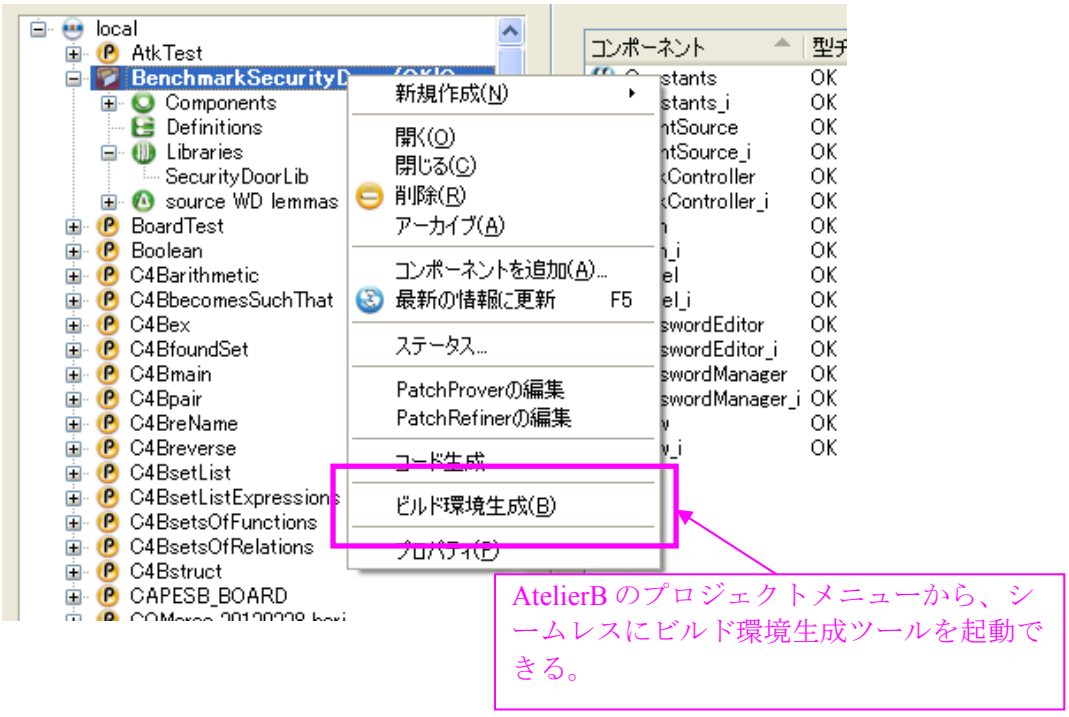


図 18　Atelier B からのビルド環境生成ツールの起動

図 19は、起動されたツールの画面である。初期状態の画面では、ツールによって解析されたBプロジェクトのモジュール依存関係が木構造として表示される。

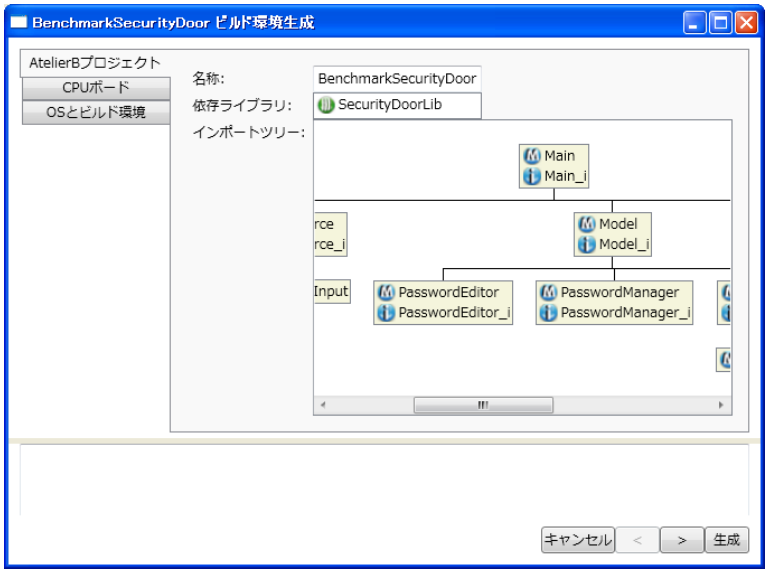


図 19　B モジュール依存関係の表示

次に、ビルド対象とするCPUボード、OS、およびビルド環境の指定をツール上で行う。図 20は、OSとビルド環境として、TOPPERS/ASPカーネルとルネサスエレクトロニクスHEWを選択した場合の画面である。

本ツールでは、サブテーマ③で扱った開発事例のような、Bプロジェクト全体が外部から周期実行される設計を前提としている。そのため、外部からメインモジュールを周期呼び出しするために必要な情報（周期実行を実現するタスク定義、駆動周期など）も、ここで併せて指定する。　これらの情報を設定した後に「生成」ボタンを押下する事で、Bコードトランスレータ(C4B)によるコード変換、および選択した環境で必要な設定ファイルやランタイムライブラリの作成が自動的に行われる。出力されたソースツリーは、対象とする開発環境で読み込み、そのままビルドする事が可能である（図 21）。

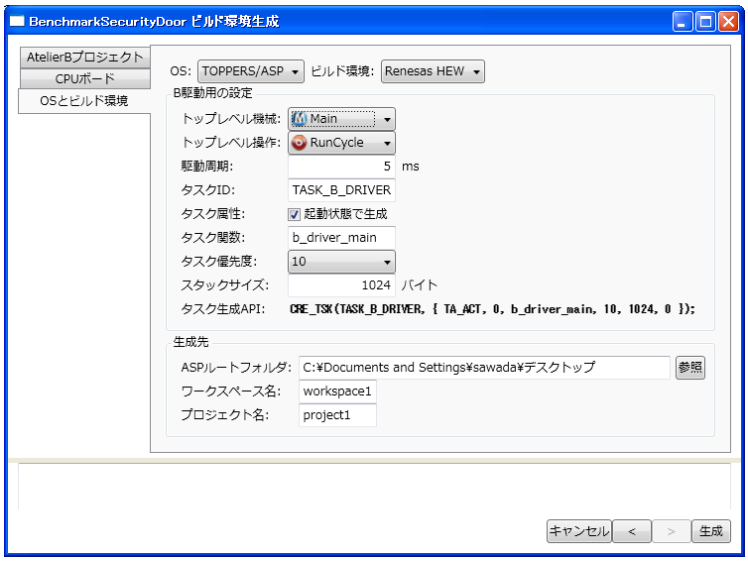


図 20 OS とビルド環境の設定画面

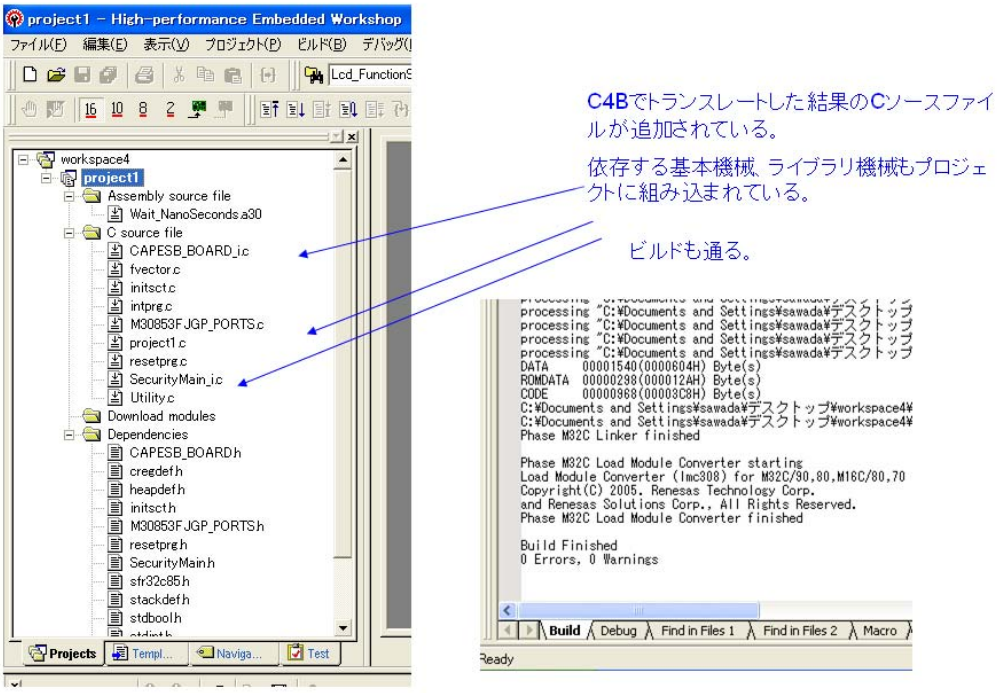


図 21　生成されたソースツリーのビルド

このように、本ツールを用いる事で、コードトランスレータのコード生成規則や、対象マイコンの開発環境に対する深い知識の習得なしに、マイコン上で実際に動作するソフトウェア開発が可能となった。これによって、川下企業等が B メソッドの導入検討のための評価試験を行う際に、本質的ではないプラットフォーム依存の煩雑な作業なしで迅速に評

価を行う事が可能となるため、B メソッドの事業化阻害要因の一つを取り除くことが出来たと考えている。

また、本ツールの構成要素の一つである B プロジェクトの解析処理部は、B プロジェクトの構造を抽象的に取り扱うための汎用クラスライブラリとして実装されている。そのため、他の補助ツール開発においてもツールの中核部分として再利用する事が可能であり、開発効率の向上に寄与する事が期待される。

２－６　　サブテーマ⑥: 形式記述教育コンテンツ開発

２－６－１ 目的と目標

サブテーマ⑥では、今までの研究で得られた形式手法に関する技術ノウハウ、要求スキルなどの知見に基づき、形式手法を導入する企業や個人向けに、技術者育成やコンサルティング事業などに利用できるＢメソッド教育コンテンツの開発を行う。

教育コンテンツのテキストは、対象とする受講者のレベルに合わせて入門編・応用編・対話証明編の内容により構成し、目的や時間などにより柔軟にカリキュラムが組めるように配慮する。また、テキストの内容に合わせた実習用課題と、動作確認用実機の制作も行い、Ｂメソッドによる組み込み機器開発の流れが体系的に理解できるような教育コンテンツとする。

また、作成した教育コンテンツを評価するため、アドバイザ企業などを対象としてパイロットセミナを実施する。セミナの結果から教育効果の測定や改善点の洗い出しを行い、今後の事業化に向けた改善を行う。

２－６－２ 研究実施内容と成果

２－６－２－（１） カリキュラムの検討と教材テキストの作成

過去２年間で蓄積してきたＢメソッドによるソフトウェア設計・開発に関する知識、知見を効果的に移転するために、入門編、応用編、対話証明編の 3 部から構成される教育カリキュラムと教材テキストを開発した。

入門編

入門編は、形式手法と B メソッドに関する事前知識を持たない技術者を対象とし、受講者が B メソッドの基本的なモデル記述と検証の作業を、支援ツールを用いながら実施できること、B メソッドの利点・特長を理解出来ること、の二点を達成目標とする構成とした。

テキストの内容は以下の通りである。

- B メソッド概要
- 数学的基礎
- B 抽象機械の記述
- リファイメント
- インプリメンテーション

応用編

応用編は、既に B メソッドの基本知識を習得している技術者を対象として、実際のソフトウェア開発業務と開発プロセス上で B メソッドをどのように活用していくのか、という観点で作成を行った。また、応用編のカリキュラムは、実習用例題として開発した「電子施錠システム」の設計プロセスを B メソッドによって用いて順に進めていく流れとした。

テキストの内容は以下の通りである。

- 電子施錠システムの要求仕様について
  - 対象の最も重要な仕様、性質は何か
  - セキュリティ仕様の形式記述

・アーキテクチャ設計
  ・B メソッドによる大規模ソフトウェアのモデリング方法
  ・アブストラクトモデル、コンクリートモデル
  ・モジュール分割とモジュール間参照

・電子施錠システムにおけるアーキテクチャ設計の事例
  ・メイン機械の記述と、モジュール分割の方針
  ・詳細化による性質の伝搬
  ・リンク不変条件の記述と、モデルの一貫性証明
・詳細設計工程
  ・コンクリートモデルの作成
  ・支援ツールを用いたコード自動生成

対話証明編

B メソッドの特色である数理的証明による仕様検証は、高い検証能力を持つ一方で、ツールによる自動証明で処理しきれない複雑な項目については人が介在する対話証明作業を行う必要がある。対話証明作業を実践するには、B メソッドの数理モデル、証明支援ツールの両方に関する高い専門知識が必要となるため、この部分を「対話証明編」として独立した教材とした。

対話証明編の教材については、サブテーマ 1 などの技術調査で作成し、プロジェクト内の技術者教育で活用してきた資料・実習課題などを取りまとめる形で作成した。

・例題: 優先度付き待ち行列モデルの証明
  ・対話証明作業の基本的な流れ
  ・証明戦略の検討
  ・Atelier B 対話証明系 の基本命令

・例題：ループを含む記述の証明
  ・ループ不変条件・ループ変条件
  ・証明を意識したループ記述

・対話証明パターン集
  ・Goal が明らかに偽となる命題の仮説否定
  ・全称命題を用いた仮説の追加
  ・存在命題の証明
  ・Technical Invariant 使用による仮説否定
  ・等式の仮説を用いた命題の置き換え

B メソッド文法早見表

基礎編、応用編ともに、早い段階からB仕様記述の例題をツール上で実際に記述していく構成となっている。しかし、教育コースの限られた時間の中では、実習作業に必要なB メソッドの文法事項を詳しく説明する事は困難である。そこで、実習作業の効率化を実現するため、実習に必要な最小限の文法事項のみを取捨選択し、参照しやすくコンパクトにまとめた「Bメソッド文法早見表」(図 22)を作成した。

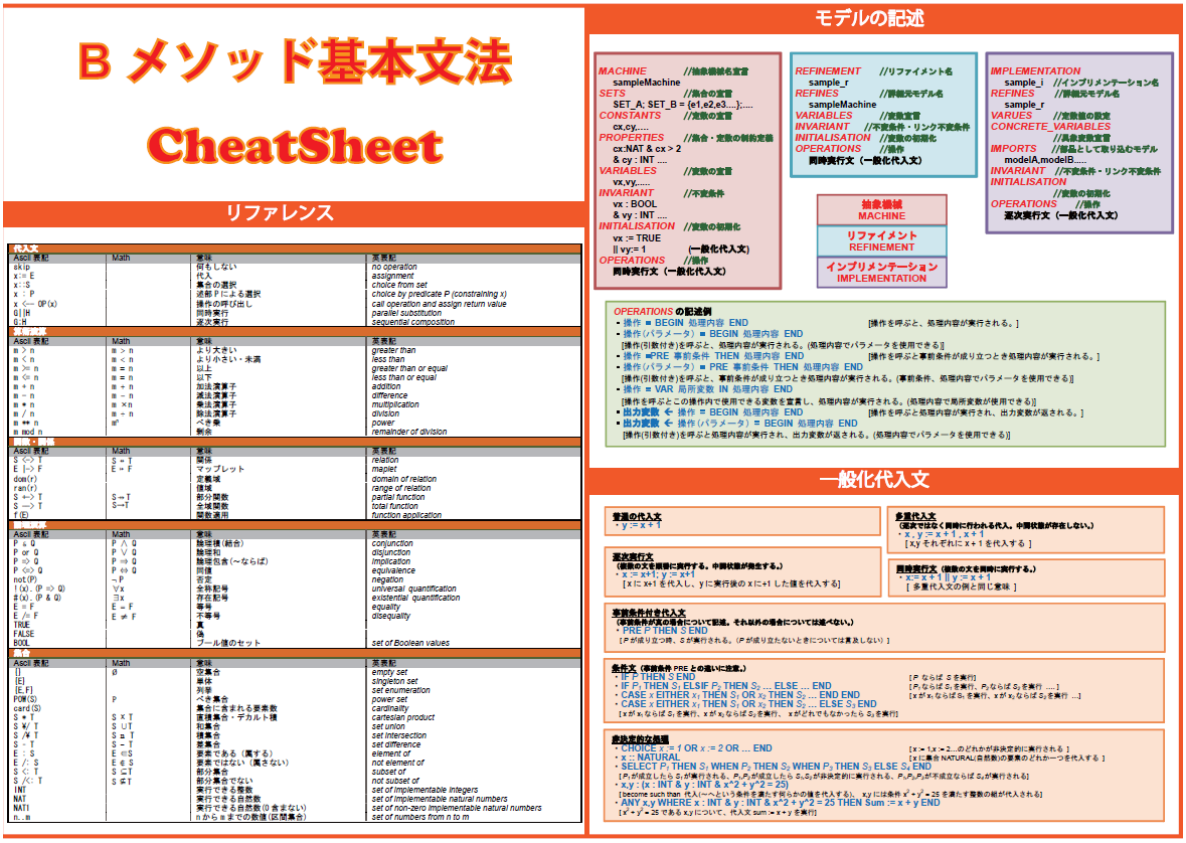


図 22　Bメソッド文法の早見表

２－６－２－（２）　実習用例題「電子施錠システム」の開発

教育コース、特に応用編で取り扱う実践的な組込システム開発の実習用例題として、「電子施錠システム」（図 23、図 24）を開発した。

図 23 電子施錠システム

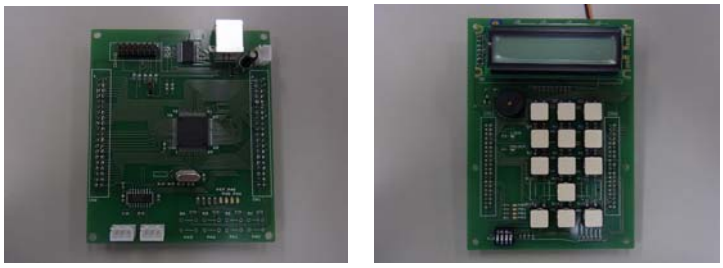


図 24　試作した教材用マイコン基板および I/O 基板

本例題は、試験導入開発の題材として扱ったドアクローザに類似した、マイコンを用いた小規模な制御システムであり、大まかには次のような仕様を持つ。

- ハードウェアの構成要素として、ドアをロックするかんぬきとそのアクチュエータ、ドア等の開閉状態を検知するスイッチ、暗証番号を入力するキーパッド、表示パネルを持つ。
- テンキーからパスワード（暗証番号）を入力することで、ドアのロックが解除される。
- ロック解除中に、パスワードの変更を行う事が出来る。

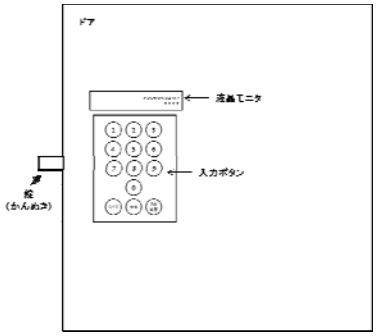


図 25　電子施錠システムのハードウェア構成

より詳細な仕様については、自然言語と状態遷移図による非形式的な「セキュリティドア要求仕様書」として作成した（図 26）。

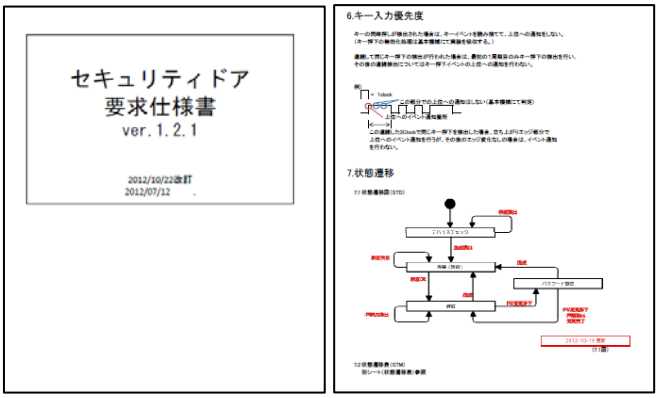


図 26　非形式的な 要求仕様書（一部）

本例題の作成にあたっては、B メソッドの特長である段階的詳細化の有用性が明解に理解出来る内容となるよう留意した。具体的には、

- システムとして満たすべき、最も基本的で重要な性質を、最初に形式的にモデリングする。
- その記述内容を出発点として段階的詳細化を行い、基本的性質の正しさを保証しながら全体の詳細設計を記述する。

という流れを、例題を通して受講者に実感出来る内容とした。。

電子施錠システムはセキュリティ関連システムであるため、今回は「正しく認証されない限りは、ドアのロックは解除されない」などといったセキュリティに関する性質に着目し、それらを表現するB抽象機械 SecurityProperties.mch (図 28)を作成した。モジュール SecurityProperties は、パスワード認証の状態(認証・未認証)、鍵の施錠状態（施錠・解錠）、およびシステムの初期化の完了状態を抽象的に表現し、また、システムがセキュアであるために必要な不変条件を記述している。

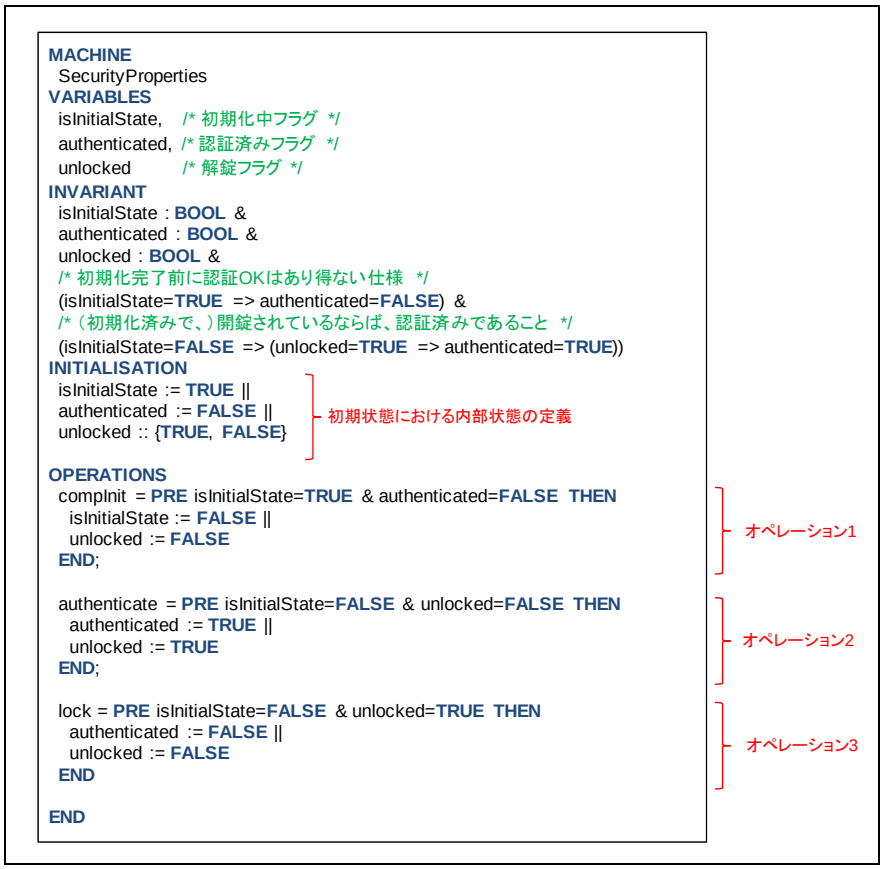


図 27　B によるセキュリティ性質のモデリング

図 28は、電子施錠システムのBアブストラクトモデルのモジュール構成である。アブストラクトモデルは、主モジュール Main を頂点に、より具体的、詳細な制御ルールや入出力動作を司るモジュールが木構造を構成している。一方、全ての詳細化の出発点となる Mainモジュールでは、システムが守るべきセキュリティ性質を記述したSecurityProperties をインクルードしている。そのため、Bメソッドの段階的詳細化の検証能力を用いること

で、最初に記述したセキュリティ性質がモデル全体にも確実に引き継がれている事を保証出来る。

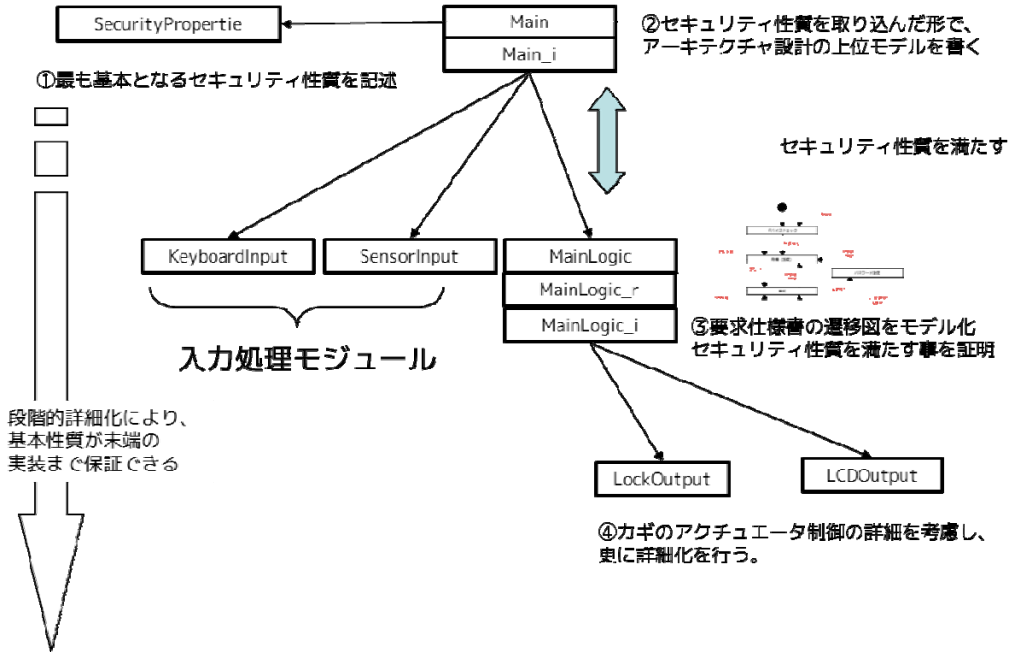


図 28　電子施錠システムの B アブストラクトモデル

２－６－２－（３）　パイロットセミナの実施

開発した教材テキスト・B モデル・ハードウェアを用いて、教育コンテンツとしての効果測定や改善提案等のフィードバックを得る目的で、アドバイザ企業等の技術者を対象としたパイロットセミナーを実施した。

- 名古屋パイロットセミナ
  - 平成 24 年 11 月 13 日　10:00～18:00
  - 参加人数：７名
  - 参加企業：自動車、半導体関連企業など

- 札幌パイロットセミナ
  - 平成 24 年 11 月 28 日 13:30～17:30、29 日 10:00～17:30（1.5 日コース）
  - 参加人数：７名
  - 参加企業：大手～中堅ソフトウェア企業など

札幌パイロットセミナにおけるカリキュラムを以下に示す。

(第 1 日)
13:30～14:30　形式手法・B メソッドの概要
14:30～17:30　モジュール仕様記述と検証
　　基本文法・数学的背景
　　簡単なモデルのモデリング実習
　　電子施錠システムセキュリティ仕様のモデリング実習
(第 2 日)
10:00～14:00　モジュールの詳細化・実装

詳細化モデル記述と検証の実習
14:00〜14:50　アーキテクチャ設計の考え方
16:00〜17:00　例題アブストラクトモデルの作成実習
17:00〜18:00　全体質疑・アンケート調査

セミナを実施した結果であるが、「基礎編」「応用編」の両方を 1 日もしくは 1.5 日に圧縮した構成であったため、全般的に時間不足であり、いずれの回でも用意した題材を全て消化しきれない結果となってしまった。今回は、B メソッドによる開発の具体的なイメージや、B メソッドの手法としての利点などを受講者に伝える事を優先し、具体的な開発例題を扱う「応用編」までを含めた構成としたが、これらの内容を一度に教えるのはやはり困難である事がわかった。

また、初級編用に用意した B 例題に関しても、我々の想定よりも難易度が高く苦戦している傾向が見られた。この点に関しては、2 回目の札幌セミナではより平易な例題を増やし演習時間を延長するなどの対策を行った結果、ある程度の改善することができた。

受講者に対するアンケートや聞き取り調査では、数理的な仕様検証が出来るという点については肯定的に評価されているものの、現在抱えている業務に対する B メソッドの適用可能性や、適用した場合のメリットについて、必ずしも十分な理解と評価が得られていない傾向が見られた。形式手法に対して受講者が抱く期待や、解決したいと望んでいる技術課題（例えば、自動車業界におけるモデルの物理的振るまいの検証など）と、B メソッド導入によって得られるメリットが必ずしも一致していないことが、この様な評価の一因と思われる。これを解決するためには、B メソッドで解決出来る問題が何であるかを、顧客ニーズに即した形で的確に伝える工夫が必要と考える。

３． 全体総括

３－１　　研究開発成果

本研究の開発成果は、サブテーマ②において ISO/IEC 15408 対応開発を途中から導入したことなど若干の変更はあるものの、概ね当初予定通りの目標水準で完了できた。

各研究項目の成果と達成度について、以下に記載する。

①　形式仕様記述手法の調査

目標達成度: 100%

得られた成果: (B メソッド教育コンテンツ等)

本研究項目は、他のサブテーマ実施の支援が目的であり、単独での成果出力の予定はない。しかし、ワーク活動などによりサブテーマ②③の試験導入開発における課題を概ね解決出来たため、当初目標を達成したものと判断した。また①で作成した調査資料等については、⑥の教育コンテンツの一部として活用されている。

②　TCP/IP プロトコルスタック開発への試験導入

目標達成度 100%

得られた成果: ISO 15408 EAL7 の形式手法関連要求事項に対応した開発成果物、外部機関によるレビュー報告書

本研究項目は、より高い水準の研究成果を得るため、当初計画を変更し、ISO 15408 規格への対応を中心に活動することとした。しかしながら、開発対象物として選定した暗号通信ミドルウェアに関する主要な設計ドキュメントおよび B による形式モデル一式を完成させ、かつ、その内容の妥当性に関して海外機関(仏 ClearSy 社)からの評価を得る事が出来た。

レビューを実施した ClearSy 社は認証機関ではないこと、また活動内容が規格の形式手法に限定されていることから、この結果からただちに「ISO 15408 EAL7 対応可能な技術水準」と主張する事は出来ないものの、B メソッドを用いた ISO 15408 対応開発に関するノウハウ、および ST(Security Target)などの ISO 15408 の基本文書の作成ノウハウは十分に取得出来たといえる。

③　自動車部品制御ソフトウェアへの試験導入

目標達成度 90%

与えられた開発対象物である「ドアクローザ」に関して、B および Event-B による設計・開発を完了し、かつ外部評価を含むテストで不具合 0 件を達成する事が出来た。また、自動コード生成により、B モデルから実際に実機上で動作するプログラムを生成し動作を確認することが出来た。また、その過程において、小規模な組込み制御システムのモデリングに関する B 記述の指針を確立することができた。これらの点については、目標通りの成果が得られたと言える。

その一方で、仕様書の内容の抽象度が非常に低く、設計に近い内容であったため、B メソッド導入による明確な効果が見えにくかった部分があった。この点に関しては、今後、他の事例において仕様作成段階から Event-B 手法の適用を試みるなどの取り組みを進める予定である。

④　形式手法の導入効果分析

目標達成度: 80%

分析対象とした 2 件の試験導入開発では、いずれも(テスト実施した範囲内で)不具合 0 件との結果が得られており、その点では当初の研究目標(各工程での不具合流出率の半減)は達成出来た。

しかし、評価対象の絶対的な件数が少ないこと、また、完全な同条件の開発における従来手法開発と形式手法導入開発の比較データは得られなかった事などがあり、アドバイザ企業等からは、より説得力のある事例データが欲しいとの要望を頂いている。この点については、今後研究メンバ機関が実施する開発業務における測定データ収集を更にすすめ、より高い精度の評価データを取得していく予定であり、サブテーマ⑥例題などを対象とした追加のデータ計測の取り組みに着手している。

⑤　形式記述支援ツール開発

目標達成度: 90%

得られた成果: 「ビルド環境作成ツール」

計画していた、μITRON, OSEK/VDX 組込み OS 環境向けの「ビルド環境作成ツール」に関連する開発を完了した。

⑥　形式記述教育コンテンツ開発

目標達成度 90%

得られた成果：教育コンテンツ（基礎、応用、対話証明）、例示課題「電子施錠システム」

予定していた三編の教育コンテンツ、および例題「電子施錠システム」開発を完了し、パイロットセミナでの評価とフィードバックを行った。教材に関しては改善を要する点も多いが、研究事業終了後のセミナや技術普及に向けて、既に教材の改訂作業を開始している。

３－２　　今後の課題と事業化計画

研究終了後の事業化活動に関しては、民間企業各社が主体となり、個別に推進していくことになる。ただし、中小企業への技術支援機関でもある管理法人（道総研）としては、各社との協力体制を維持しながら事業化に関する活動を支援する予定である。

一部の研究メンバ企業は、セキュアな通信ミドルウェア開発に関する新たな研究課題を既に立ち上げている。その中の要素技術として本研究における形式手法関連技術の活用進められており、その延長上としての事業化が期待される。

サブテーマ⑥教育コンテンツに関しては、道総研と研究メンバ企業の連携により、技術セミナ等の開催を積極的に行い、教育のビジネス化や、川下企業に対するコンサルティング業務の獲得などを目指す。

サブテーマ⑤形式記述導入支援ツールに関しては、ただちに有償製品として販売する事は難しいが、市場での認知度を高めるために基本部分のオープンソース化なども視野に入れた戦略を検討していく。