

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

お客様各位

資料中の「日立製作所」、「日立XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って三菱電機株式会社及び株式会社日立製作所のマイコン、ロジック、アナログ、ディスクリート半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。従いまして、本資料中には「日立製作所」、「株式会社日立製作所」、「日立半導体」、「日立XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

ルネサステクノロジ ホームページ (<http://www.renesas.com>)

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

ご注意

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。

Ho7000シリーズ

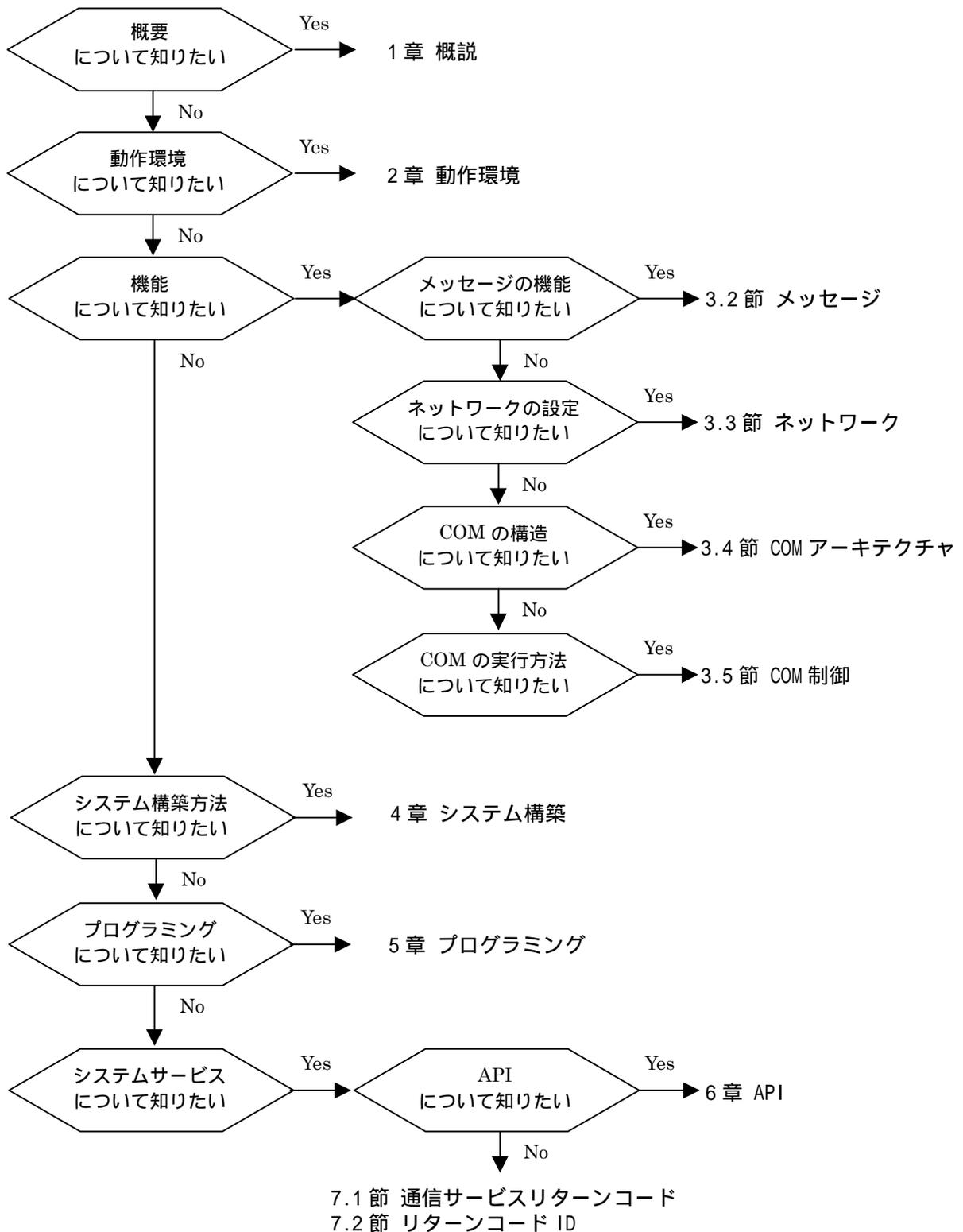
通信マニュアル

ご注意

1. 本製品(ソフトウェア製品及びその関連ソフトウェア製品を含む。以下、同じ。)の使用に際しては、「外国為替及び外国貿易法」等、技術輸出に関する日本及び関連諸国の関係法規の遵守が必要となります。
2. 弊社は、本製品の使用に際しては、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等の権利に関し、別途、個別の契約書等(マニュアルの記載を含む。以下、同じ。)にて弊社による明示的な許諾がある場合を除き、その保証または実施権の許諾を行うものではありません。また本製品を使用したことにより第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
3. 本製品およびその仕様、またはマニュアルに記載されている事柄については、将来、事前の予告なしに変更することがありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書(マニュアルを含む)をご確認ください。
4. 本製品の使用(マニュアル記載事項に基づくものも含む)により直接または間接に生ずるいかなる損害についても、弊社は一切の責任を負いません。また、本製品の配布に使用される搭載機器や媒体が原因の損害に対しましても、弊社は一切の責任を負いません。
5. 本製品を、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途向けには使用できません。お客様の用途がこれに該当するかどうか疑問のある場合には、事前に弊社営業担当迄ご相談をお願い致します。
6. 本製品を使用してお客様のシステム製品を設計される際には、通常予測される故障発生率、故障モードをご考慮の上、本製品の動作が原因での事故、その他の拡大損害を生じないようにフェールセーフ等の十分なシステム上の対策を講じて頂きますようお願い致します。
7. 本製品およびマニュアルの著作権は弊社が所有しております。お客様は、弊社から提供された本製品を、別途、個別の契約書等にて定める場合を除き、いかなる場合においても全体的または部分的に複写・解析・改変することはできないものとします。
8. お客様は、別途、個別の契約書等にて定める場合を除き、本製品のマニュアルの一部または全部を無断で使用、複製することはできません。
9. 弊社は、本製品を1台のコンピュータで使用する権利をお客様に対してのみ許諾します。よって、本製品を第三者へ譲渡、貸与、賃借することは許諾しないものとします。但し、別途、個別の契約書等にて定められる場合はその条件に従います。

10. 本製品をはじめ弊社半導体およびその関連製品についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

マニュアルを読まれる前に、知りたい事項を下記フローからピックアップされることをおすすめします。



目次

1	概説	1
1.1	はじめに	1
1.2	略語	1
1.3	概要	2
1.4	本製品の特長	3
2	動作環境	4
2.1	準拠バージョン	4
2.2	コンフォーマンスクラス	4
2.3	通信バス	4
2.4	オペレーティングシステム	5
2.5	注意事項	5
3	機能	6
3.1	機能概要	6
3.2	メッセージ	6
3.2.1	メッセージ種類	6
3.2.2	メッセージ操作	7
3.2.3	アプリケーションへの通知	8
3.2.4	送信モード	9
3.2.5	タイムアウト監視	13
3.2.6	メッセージ属性	14
3.3	ネットワーク	15
3.3.1	HCAN	15
3.3.2	チャンネル	16
3.4	COM アーキテクチャ	17
3.5	COM 制御	18
3.5.1	起動	19
3.5.2	COM 動作	19
3.5.3	終了	19
3.5.4	エラー処理	20
4	システム構築	22
4.1	COM アプリケーションの構築	22

4.2	COM 構築ファイル	23
4.2.1	Evb	24
4.2.2	Hcan	24
4.2.3	Mcs	25
4.2.4	Msg	25
5	プログラミング	26
5.1	COM 初期化	26
5.2	インクルードファイル	26
5.3	割り込み	27
5.3.1	HCAN 割り込み	27
5.3.2	割り込み優先レベル	28
5.4	ユーザコード	28
5.4.1	初期化处理	28
5.4.2	コールバック関数	28
5.4.3	HCAN ISR	30
5.5	OS とのインタフェース	31
5.5.1	割り込み	31
5.5.2	アラームとタスク	31
5.6	レジスタ	31
5.7	スタック	31
5.8	アセンブラルーチンからの通信サービスコール	32
5.9	アセンブラ使用時の注意事項	32
6	API	33
6.1	標準インタフェース	33
6.1.1	StartCOM	33
6.1.2	MessageInit	34
6.1.3	SendMessage	35
6.1.4	ReceiveMessage	36
6.1.5	GetMessageStatus	37
6.2	独自インタフェース	38
6.2.1	OC_SendMsgToNetPer	38
6.2.2	OC_MixedTxEval	39
7	付録	40
7.1	通信サービスリターンコード	40
7.2	リターンコード ID	41
7.3	通信サービス呼び出し	42

7.4	データ型.....	42
7.5	制限値.....	43

1 概説

1.1 はじめに

本書は、OSEK/VDX(以下、OSEK と略します)仕様 通信 バージョン 2.0a に準拠した Ho7000 シリーズの通信 V1(以下、COM と略します)について記述します。本書は OSEK 仕様を理解していることを前提として記述しています。

本製品をご使用になる前に本書をよく読んで理解してください。また、次の関連ドキュメントもお読みの上、理解してください。

- ・ “OSEK/VDX Operating System”, Version 2.0 revision 1 (OSEK/VDX 運営委員会)
- ・ “OSEK/VDX Communication”, Version 2.0a (OSEK/VDX 運営委員会)
- ・ “OSEK/VDX Communication”, Version 2.1 revision 1 (OSEK/VDX 運営委員会)
- ・ 本製品のリリースノート
- ・ SuperH RISC engine C/C++コンパイラパッケージマニュアル
- ・ 使用する SH マイコンのハードウェアマニュアル、プログラミングマニュアル

1.2 略語

API	アプリケーションプログラムインタフェース
CAN	コントローラエリアネットワーク
CCC	通信コンフォーマンスクラス
COM	日立自動車向けオペレーティングシステム 通信
DLL	データリンク層
ECU	電子制御ユニット
HCAN	日立コントローラエリアネットワーク または 日立コントローラエリアネットワーク 2
HW	ハードウェア
NM	ネットワーク管理
OS	オペレーティングシステム
OSEK	オープンシステムおよび自動車エレクトロニクス用通信インターフェース
SW	ソフトウェア
VDX	自動車分散制御

1.3 概要

COM は、OSEK 仕様によって規定されている SW の中の 1 つです。その他に、オペレーティングシステム (OS) とネットワーク管理 (NM)があります。3 つの SW はいずれも、自動車用アプリケーションに共通の機能を提供します。

図 1.1は自動車用アプリケーションのシステム概念を示しています。自動車には複数の ECU から構成され、ネットワーク上で情報を交換します。例えば ECU にはエンジン制御、ギアボックス制御などがあります。より高度な機能は、複数の ECU で構成されることもあります。

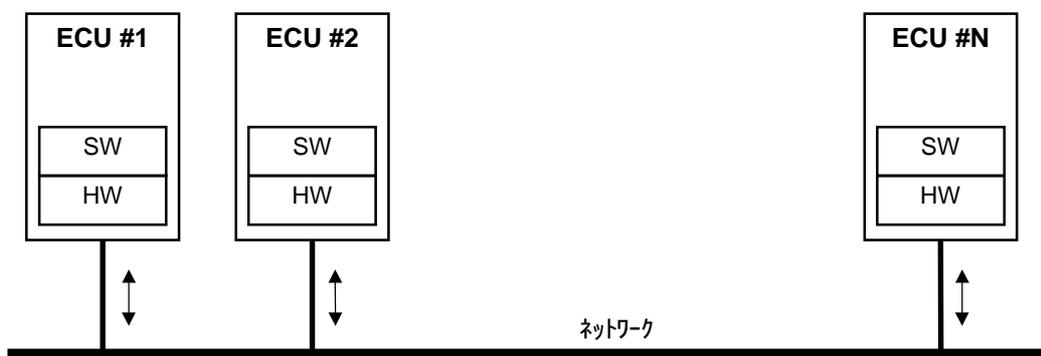


図 1.1 自動車用アプリケーションのシステム概念

図 1.2に ECU の内部構造を示します。

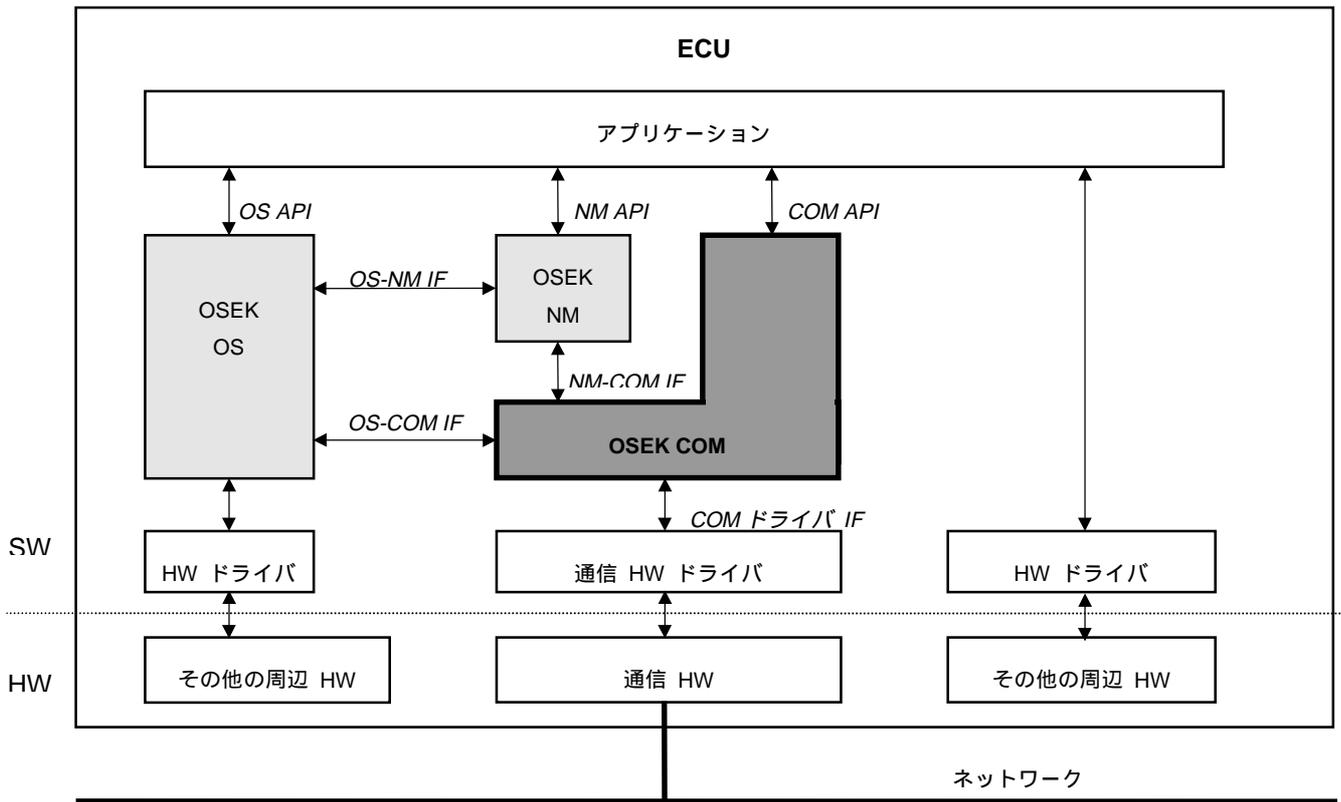


図 1.2 ECU 内の SW 構造

COM は ECU 内、および ECU 間通信を制御します。また、メッセージ送受信のサービスを提供します。ECU 内および ECU 間におけるメッセージ送受信サービスの API は同じインタフェースです。

COM には上位互換性を持った通信コンフォーマンスクラス (CCC)があります。本製品では CCC0 および CCC1 のコンフォーマンスクラスをサポートします。CCC0 は最小システムで構成され、OS を組み込まない状態でのインプリメントが可能です。CCC1 は OS が必要です。各 CCC は、各アプリケーションの通信性能や要求に適應できるように準備されています。

COM は、デバイスドライバを利用して、通信 HW へアクセスします。このインタフェースは OSEK 仕様によって規定されていません。本製品では CAN デバイスを使用します。

1.4 本製品の特長

以下に、本製品の特長を示します。

- COM は、ANSI C 仕様で定義されたアプリケーションプログラムインタフェースの提供によりアプリケーションの移植性を支援します。
- COM は、次の通信機能を提供します。
 - タスク間通信
 - ECU 間通信
 - 送信方式：直接送信/周期送信/混合送信
 - 通信タイムアウト監視
 - イベントセットとタスク起動による通知

【注】送信方式の周期送信、混合送信は独自インタフェースを提供します。詳細は3.2.4 送信モードを参照してください。

2 動作環境

2.1 準拠バージョン

COM は OSEK/VDX 通信バージョン 2.0a に準拠しています。さらに、バージョン 2.1 リビジョン 1 に準拠したメッセージ送信モード、タイムアウト監視をサポートしています。

2.2 コンフォーマンスクラス

表 2.1 に COM のコンフォーマンスクラスに対応する機能を✓で示します。

表 2.1 コンフォーマンスクラス

機能、サービス		コンフォーマンスクラス	
		CCC0	CCC1
送信概念	直接送信	✓	✓
	周期送信		✓
	混合送信		✓
通信タイムアウト監視			✓
メッセージ	キューなし	✓	✓
プロトコル	UUDT (非分割メッセージ)	✓	✓
サービス	SendMessage	✓	✓
	ReceiveMessage	✓	✓
	GetMessageStatus	✓	✓
通知	タスク起動		✓
	イベントセット		✓

CCC0

ECU 内および ECU 間通信に必要な最低限のサービスと機能を提供します。OS は必須ではありません。

CCC1

周期送信、混合送信、タイムアウト監視、通知機能を提供します。OS が必須です。

2.3 通信バス

COM は、CAN バス (以下、バスと略します) 上で動作します。通信 HW は HCAN を使用します。その他の通信 HW およびバス上では動作しません。

2.4 オペレーティングシステム

COM は、該当する「Ho7000 シリーズ」の OS と組み合わせて使用することができます。その他の OS と組み合わせて使用することはできません。

2.5 注意事項

- (1) COM は、CPU 内蔵 HCAN 用ドライバを提供します。このドライバは OSEK 通信仕様には含まれていませんが、COM の動作に必要ですので削除しないでください。
- (2) COM の構築については COM コンフィギュレータのヘルプファイルを参照してください。
- (3) リトルエンディアンでは使用できません。ビッグエンディアンで使用してください。

3 機能

3.1 機能概要

図 3.1に COM の機能の概要を示します。

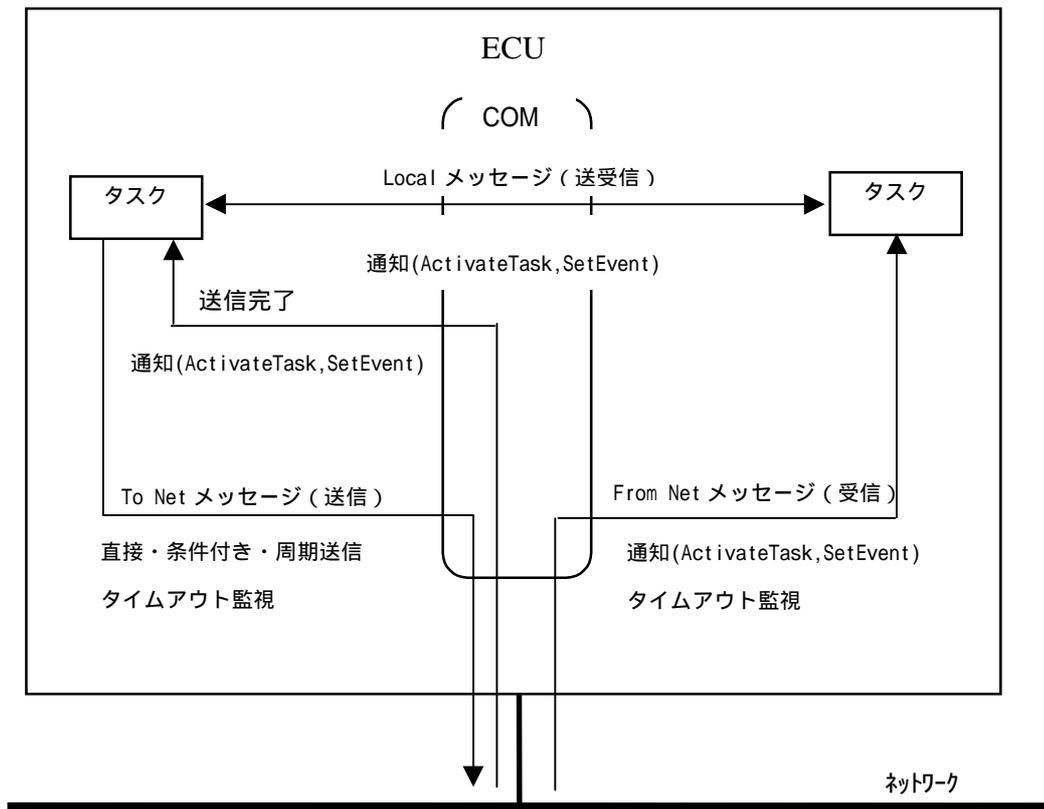


図 3.1 COM 機能概要

3.2 メッセージ

3.2.1 メッセージ種類

次の3種類のメッセージがあります。

- Local (ローカル) メッセージ
- To Net (ネットへ送信する) メッセージ
- From Net (ネットから受信する) メッセージ

(1) Local

Local メッセージは ECU 内で送受信されるメッセージです。ネットワーク上に送信されることはありません。これらのメッセージは、単一アプリケーション内での通信で使用します。

(2) To Net

To Net メッセージは、ECU からネットワークへ送信されるメッセージです。このメッセージは、他の ECU によってネットワークから受信することができます。このメッセージは送信側の ECU によってローカルに受信することも可能です。

(3) From Net

From Net メッセージは、ネットワークから ECU に受信されるメッセージです。他の ECU からネットワークへ送信されたメッセージを受け取ります。

3.2.2 メッセージ操作

次の操作が可能です。

- メッセージの送信
- メッセージの受信
- メッセージの状態参照

(1) 送信

メッセージは SendMessage 通信サービスを介して送信されます。送信の処理内容はメッセージ種類により異なります。

Local

Local メッセージの送信は、メッセージデータをアプリケーションから COM 内メッセージデータ領域へコピーする処理です。

To Net

To Net メッセージの場合、Local メッセージと同様にメッセージデータはアプリケーションから COM 内メッセージデータ領域にコピーされ、メッセージデータはネットワークへ送信されます。

上記は、直接送信モードを使用した場合です。送信モードの詳細については、3.2.4節を参照してください。

(2) 受信

Local

このサービスは COM 内メッセージデータ領域からアプリケーションへメッセージデータをコピーします。

From Net

バスからメッセージを受信したとき、その時点ですぐに COM 内メッセージデータ領域が更新されます。

ReceiveMessage 通信サービス呼び出し時、メッセージデータが COM 内メッセージデータ領域からアプリケーションへコピーされます。**バスからの受信は、ReceiveMessage 通信サービス呼び出しとは非同期に処理されま**
す。

(3) 状態参照

メッセージの状態は、GetMessageStatus 通信サービスを介して得られます。この命令は全てのメッセージ種類で有効です。状態は下記の通りです。

- メッセージ使用状態: ロック中 (使用中)、ロック解除 (使用していない)
- メッセージデータ値: 一回以上更新済み、未更新

メッセージ使用状態

COM 内部でメッセージを処理している間はロック中となります。ロック中、アプリケーションがメッセージにアクセスすることは出来ません。たとえアプリケーションが通信サービスを発行しても、メッセージは更新されず、ロック中を示すエラーコードが返ります。メッセージが使用できる時は、状態がロック解除となっているときです。

メッセージデータ値

COM 起動時は、全てのメッセージの状態は「未更新」になっています。一度でもアプリケーションから送信またはネットワークから受信されたとき、「一回以上更新済み」に変わります。以降、「未更新」になることはありません。

3.2.3 アプリケーションへの通知

本機能は CCC1 のみ使用可能です。メッセージの操作は、アプリケーションとは非同期に行われます。ネットワークへの送信はアプリケーションが要求しますが、送信処理は通信サービスがアプリケーションに復帰したときに完了していません。CCC1 の場合、通知機能により、送信が完了したことを COM からアプリケーションへ通知することが可能です。ネットワークから受信するとき、アプリケーションは介入できません。COM はバスから新しいメッセージを受け取り、それを COM 内部のメッセージデータ領域に格納します。アプリケーションへの通知は、この処理が完了すると行われます。これを表 3.1 に示します。

表 3.1 通知タイミング

処理	タイミング	該当メッセージ種類
メッセージ送信	SendMessage 発行時	Local
	バスへのメッセージの送信が完了した時	To Net
メッセージ受信	バスからメッセージを受信完了した時	From Net

通知方法は 2 通りあります。

- COM がイベントをセットする。
- COM がタスクを起動する。

3.2.4 送信モード

COM には下記の送信モードがあります。

- 直接送信
- 周期送信 (CCC1 のみ)
- 混合送信 (CCC1 のみ)

表 3.2 メッセージ送信モード

直接送信	送信メッセージは、SendMessage 通信サービスの発行によって無条件に送信されます。
周期送信 (CCC1 のみ)	送信メッセージは、周期アラームのタスク起動によって送信されます。
混合送信 (CCC1 のみ)	送信メッセージは、周期アラームのタスク起動によって送信されます。さらに、条件によっても送信されます。この条件はユーザが定義します。

(1) 直接送信

送信メッセージは、SendMessage 通信サービスの発行によって無条件に送信されます。直接送信は、Local と To Net のメッセージ種類で有効です。

(2) 周期送信

CCC1 を使用した場合、OS と組み合わせることにより、アラームを利用した周期送信が行えます。送信メッセージは、周期的に送信されます。このモードでは、SendMessage 通信サービスはネットへの送信を開始しません。メッセージデータ更新のみ行います。

ネットへの送信は、周期的に起動されるタスクによって開始されます。タスクおよびアラームはユーザが定義する必要があります。

周期タスクは独自の送信サービスを介して、送信を開始します。このサービスの名称は OC_SendMsgToNetPer 通信サービスです。周期タスクからこの通信サービスを呼び出してください。詳細については6章を参照してください。

COM 初期化が完了するまで、周期送信を開始しないでください。周期送信は COM 初期化後 (MessageInit()関数以降) に開始してください。

周期送信は、To Net メッセージ種類で有効です。

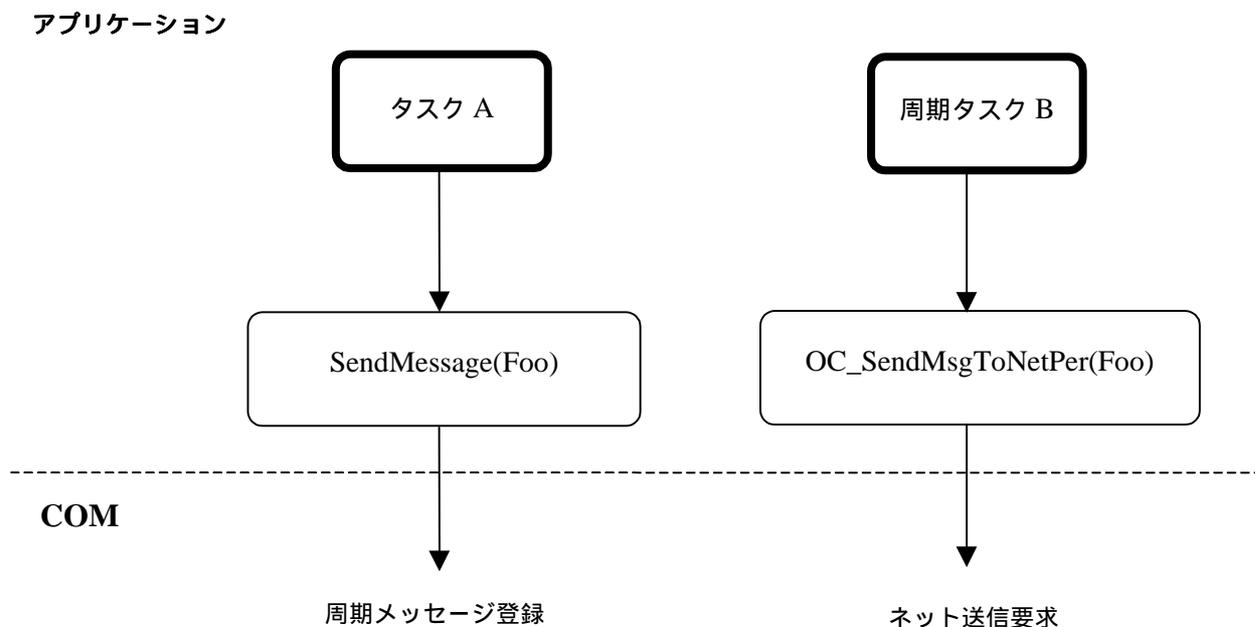


図 3.2 周期送信

タスク A は周期タスク B に依存することなく実行されます。タスク B は起動される度にネットワークへ送信を要求します。Foo という名称のメッセージが周期送信モード用に構築されています。

(3) 混合送信

CCC1 を使用した場合、OS と組み合わせることにより、アラームを利用した混合送信が行えます。送信メッセージは、周期的に送信されます。さらに、SendMessage 通信サービスによりネットワークへの送信を条件付で開始することが可能です。条件は、SendMessage 通信サービスが呼び出されるたびに評価されます。条件が満たされれば送信が開始され、そうでない場合、送信は開始されません。

このモードでは、SendMessage 通信サービスが、

1. メッセージデータを更新します。
2. 条件判定のための独自関数 OC_MixedTxEval() を呼び出します。
3. 条件が満たされた場合、ネットワークへの送信を開始します。

OC_MixedTxEval() 関数で行われる評価はアプリケーションによって定義してください。また、メッセージ送信あるいは非送信を関数のリターン値として返してください。詳細については6章を参照してください。

ネットへの送信は、周期的に起動されるタスクによって開始されます。タスクおよびアラームはユーザが定義する必要があります。

周期タスクは独自の OC_SendMsgToNetPer 通信サービスを介して、送信を開始します。周期タスクからこの通信サービスを呼び出してください。

COM 初期化が完了するまで、周期送信を開始しないでください。周期送信は COM 初期化後 (MessageInit() 関数以降) に開始してください。

混合送信は、To Net メッセージ種類で有効です。

アプリケーション

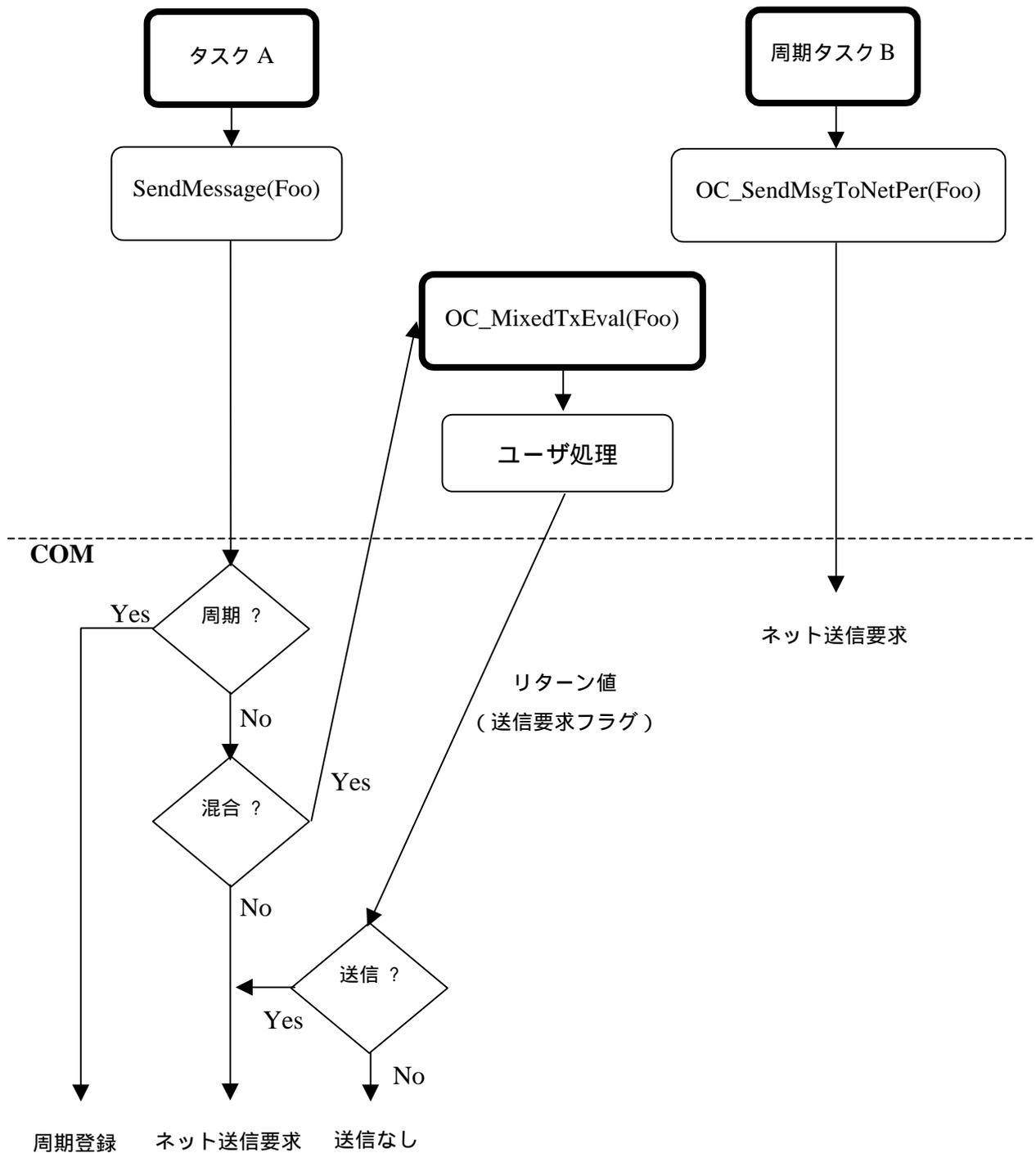


図 3.3 混合送信

タスク A は周期タスク B に依存することなく実行されます。タスク A は SendMessage 通信サービスを発行し、条件付ネットワーク送信を開始します。タスク B は、起動される度にネットワークへ送信を開始します。Foo という名称のメッセージが混合送信モード用に構築されています。

3.2.5 タイムアウト監視

CCC1 を使用した場合、OS と組み合わせることにより、アラームを利用したタイムアウト監視が行えます。タイムアウト監視とは、期限切れを監視する機能です。これは、メッセージの送信および受信どちらでも使用できます。送信では、ネットワークへ送信開始後、ある一定時間内に送信完了しているかどうかを監視します。この時間枠を超過するとアラームが満了し、タスク起動あるいはイベントがセットされます。受信では、ある一定時間内に受信完了しているかどうかを監視します。この時間は、前に受信したメッセージの受け取り完了と同時に設定されます。時間を超過した場合、アラームが満了し、タスク起動あるいはイベントがセットされます。

アラーム、タスク、およびイベントはユーザが定義する必要があります。

タイムアウト監視は To Net または From Net メッセージで使用可能です。

(1) 送信

COM がネットワークへ送信を開始する直前にアラームが開始します。これは次のサービスで行われます。

- SendMessage 通信サービス
- OC_SendMsgToNetPer 通信サービス

送信完了するとアラームは自動的にキャンセルされます。制限時間内に送信完了しない場合、アラームは満了します。

送信のタイムアウト監視はすべての送信モードで使用可能です。

(2) 受信

受信のタイムアウト監視の開始は、タスクからアラームを開始するか、あるいは一度タイムアウト監視を指定したメッセージを受信することにより可能です。受信のタイムアウト監視では、アラームはキャンセルされません。アラームは自動的に再開始されるだけです。

COM 初期化が完了するまで、受信タイムアウト監視を開始しないでください。MessageInit()関数以降から開始してください。

ネットワークからメッセージを受信する度にアラームが再開始されます。制限時間内に受信完了しない場合、アラームは満了します。

3.2.6 メッセージ属性

各メッセージに以下の属性を指定してください。これらは、COM コンフィギュレータ上で指定可能です。詳細についてはCOM コンフィギュレータのヘルプファイルを参照してください。

- メッセージ名称（英字またはアンダラインで始まり、そのあとに0個以上の英字、アンダラインまたは数字が続く文字列。最大32文字）
- データ長（最大8バイト）
- メッセージ種類
 - Local
 - To Net
 - From Net
- 通知機能
 - 無し
 - イベントマスク（イベント設定）
 - タスク ID（タスク起動またはイベント設定のためのタスク ID）
- 送信モード
 - 直接送信
 - 周期送信（CCC1のみ）
 - 混合送信（CCC1のみ）
- タイムアウト監視
 - アラーム ID
 - 満了時間
- HCAN
 - メールボックス番号（0~15）
 - CAN ID ビット長（11ビットまたは29ビット）
 - CAN ID

以下の点にご注意ください。

- (1) 同一名称のメッセージを複数設定しないでください。
- (2) 同一 CAN ID を複数設定しないでください。
- (3) メールボックス番号 16 以上は使用できません。

3.3 ネットワーク

3.3.1 HCAN

各ネットメッセージには、HCANに関連した以下の設定があります。

- メールボックス
- CAN ID

(1) メールボックス

メッセージとネットワークの間でのデータ処理は、メールボックスを介して行われます。メールボックスはメッセージデータとメッセージ制御情報を持つ HCAN 内のデータ領域です。各ネットメッセージは、必ずメールボックスを使用します。

次のメールボックスが HCAN に存在します。

- 送信用または受信用メールボックス (メールボックス 1~15)
- 受信専用メールボックス (メールボックス 0)

メールボックス 1~15

これらのメールボックスは To Net または From Net のメッセージに使用することが可能です。複数のメッセージを同一メールボックスに割り当てることはできません。

メールボックス 0

このメールボックスは From Net メッセージのみ使用することが可能です。複数のメッセージを割り当てることができます。受信処理中、メールボックス 0 での複数メッセージ受信はエラーとなり処理されません。

なお、メールボックス番号 16 以上は使用できません。

(2) CAN ID

各メッセージは、CAN バス上のメッセージ識別のために固有の識別子(CAN ID)を指定する必要があります。CAN ID は 11 ビットまたは 29 ビットで構成されています。11 ビットは、11 ビットの標準 ID が CAN バスフレーム内で使用されます。29 ビットは、18 ビットの拡張 ID および 11 ビットの標準 ID が使用されます。

3.3.2 チャネル

COMは、1つのECUで1つあるいは2つのHCANを扱うことができます。HCANには以下の設定があります。これらは、COMコンフィギュレータ上で指定可能です。詳細はハードウェアマニュアルおよびCOMコンフィギュレータのヘルプファイルを参照してください。

- メールボックス0
 - CAN ID ビット長 (11 ビットまたは 29 ビット)
 - CAN ID
 - LAFM (メールボックス0専用ローカルアクセプタンスフィルタマスク)
- 方式設定
 - 送信優先度 (メールボックス順または ID 順)
 - ビットサンプルポイント (1点または3点)
- ビットコンフィギュレーション
 - ボーレートプリスケアラ (BRP)
 - タイムセグメント1 (TSEG1)
 - タイムセグメント2 (TSEG2)
 - 最大ビット同期幅 (SJW)
- 割込み
 - 各種 HCAN 割込みの有効、無効

ビットコンフィギュレーションの設定の制限を以下に示します。

$TSEG1 > TSEG2 \geq SJW$ (SJW = 1 ~ 4)
 $3 + TSEG1 + TSEG2 = 8 \sim 25$ Time Quanta
 $TSEG2 > B'001$ (BRP = B'000000)
 $TSEG2 > B'000$ (BRP > B'000000)

詳細はハードウェアマニュアルを参照してください。

3.4 COMアーキテクチャ

COM は大きく分けて以下の 2 つの部分で構成されています。

- メッセージパッケージ
- ネットパッケージ

メッセージパッケージは COM API を提供します。ネットパッケージは HCAN ネットワークの送受信を行います。ECU 内の Local メッセージはネットパッケージを使用せず、メッセージパッケージのみを使用します。To Net および From Net メッセージは、両方のパッケージを必要とします。

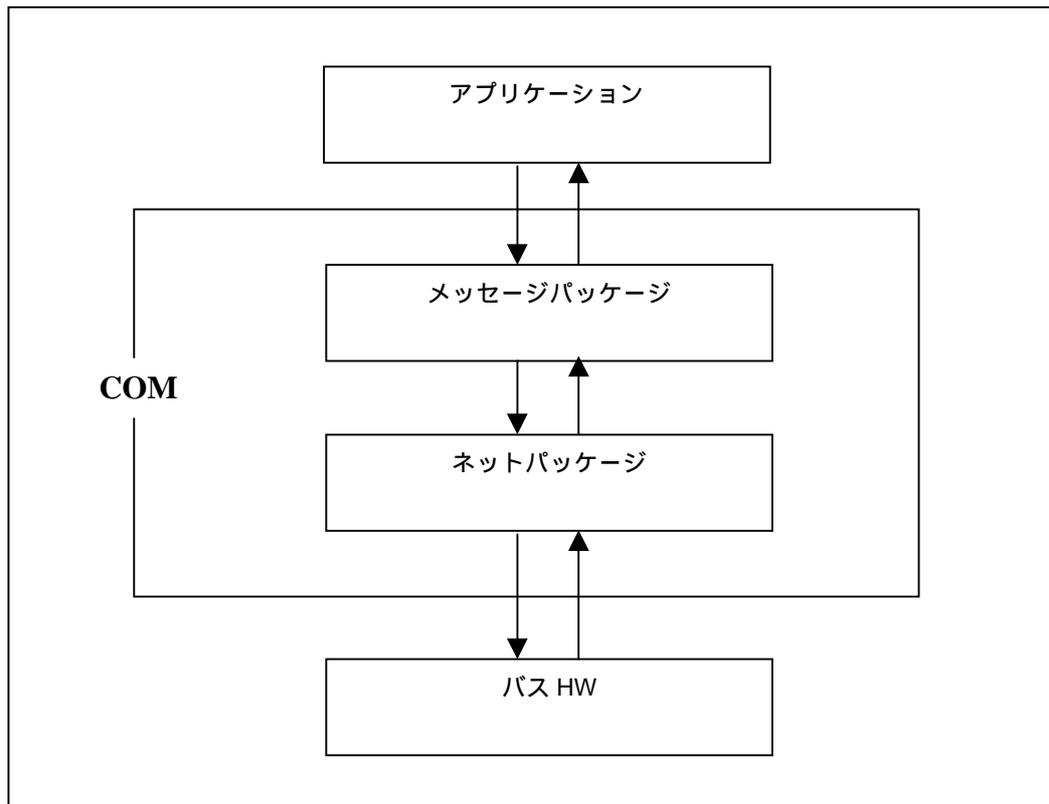


図 3.4 COM アーキテクチャ

3.5 COM制御

COMの処理手順を以下に示します。

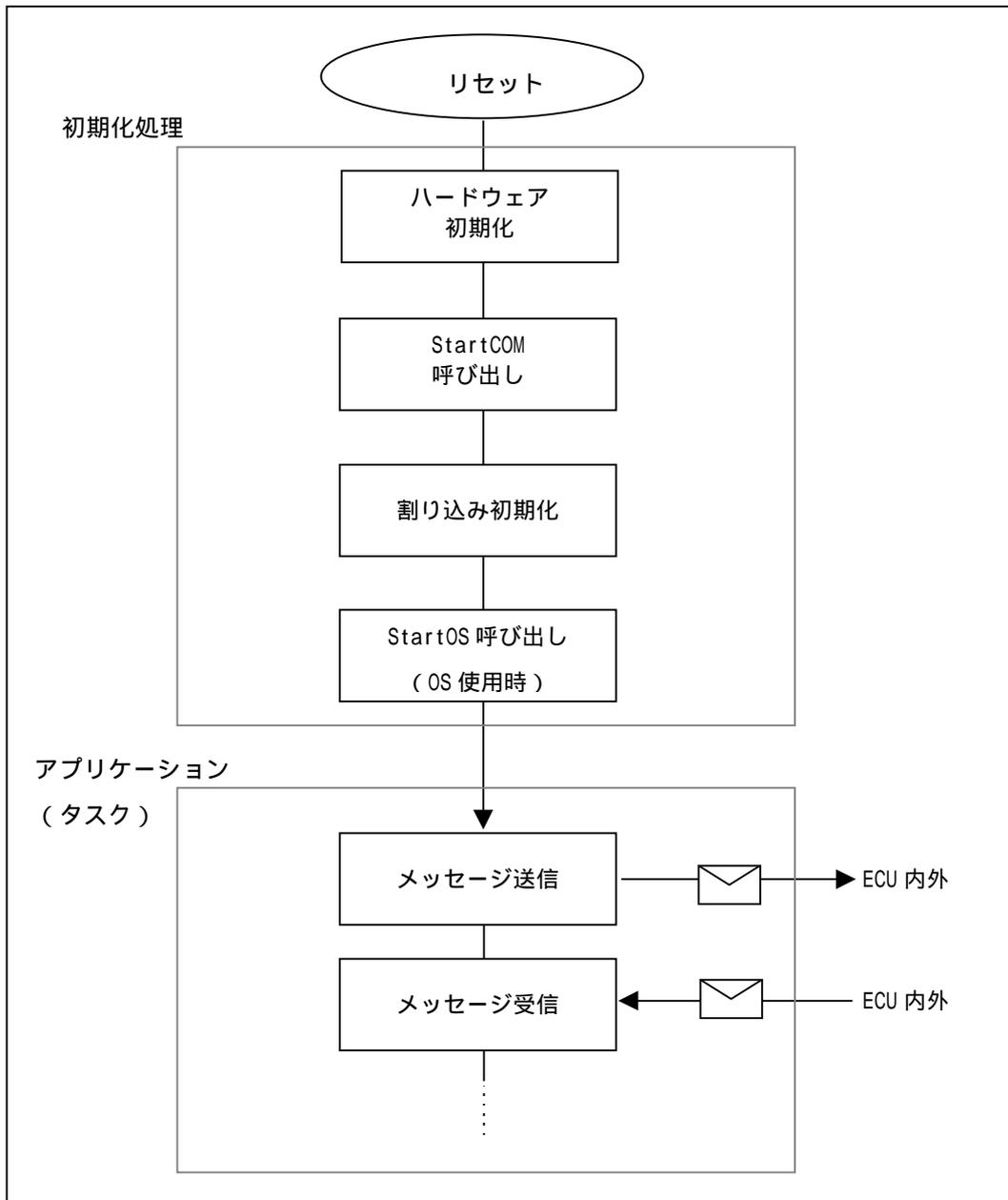


図 3.5 COM 処理手順

3.5.1 起動

COMの起動方法を下記に示します。

1. HWリセット
2. 全割り込みマスク状態でのHW初期化
3. StartCOM通信サービス呼び出し
4. 割り込み初期化
5. OSを使用している場合、StartOSシステムサービス呼び出し

上記2、4のコードをアプリケーションで用意してください。本製品は、それらのコードをサンプル提供しています(4.2節参照)。

起動中、メッセージを送受信しないでください。

3.5.2 COM動作

起動完了後、COMは通常動作に入ります。この状態でメッセージ送受信を行うことができます。送受信の流れを以下に示します。

(1) メッセージ送信

1. アプリケーションがSendMessageサービスを呼び出します。
2. SendMessageサービスが、アプリケーションからCOM内データ領域にメッセージデータをコピーします。
3. ネットワークメッセージの場合、SendMessageサービスがネット送信を開始します。
4. SendMessageサービスが復帰します。
5. メッセージがHCANからCANバスへ送られます。
6. 送信が完了すると、メールボックス空き割り込みが発生します。
7. 割り込みサービスルーチン(以降、ISRと略します)がCOMのメッセージパッケージとネットパッケージを呼び出します。この処理によってCOMが送信完了を認識します。
8. ISRが復帰します。

(2) アプリケーション内でのメッセージ受信

1. アプリケーションがReceiveMessageサービスを呼び出します。
2. ReceiveMessageサービスが、COM内データ領域からアプリケーションにメッセージデータをコピーします。
3. ReceiveMessageサービスが復帰します。

(3) ネットワークからのメッセージ受信

1. バスからメッセージが受信されると、メッセージ受信割り込みが発生します。
2. ISRがCOMのメッセージパッケージとネットパッケージを呼び出します。
3. COMは、HCANからCOM内データ領域にメッセージデータをコピーします。
4. ISRが復帰します。

3.5.3 終了

COMを終了することはできません。COMを初期化する場合は、StartCOM通信サービスを発行してください。

3.5.4 エラー処理

以下のエラーがあります。

- 通信サービスから復帰したときのエラーコード
- HCAN 関連のエラー割り込み
- 例外

(1) エラーコード

二つのエラーモードから選択することが可能です。

- 標準ステータスモード
- 拡張ステータスモード

拡張ステータスはアプリケーションの開発、デバッグに有効です。標準ステータスはデバッグされたシステムに有効です。通信サービスによって返されるエラーコードは7.1節を参照してください。

(2) エラー割り込み

HCAN には以下の割り込みがあります。

表 3.3 HCAN 割り込み

割り込み原因	初期設定	コンフィギュレータ設定
受信メッセージ	有効 (COM 予約)	不可
未読メール	有効 (COM 予約。ユーザコード追加可)	不可
送信メールボックス空	有効 (COM 予約)	不可
リセット	有効 (ユーザ変更可)	不可
オーバーロードフレーム	無効	可
バスオフ	無効	可
エラーパッシブ	無効	可
バス受信オーバーロードワーニング	無効	可
バス送信オーバーロードワーニング	無効	可
リモートフレーム要求	無効	可
バス動作要求	無効	可

COM で予約されていない割り込みは全て、アプリケーションで使用することができます。COM は、これら全ての割り込みに対してデフォルトのハンドラを提供します。必要に応じてコードを追加してください。詳細は5.3節を参照してください。

(3) 例外

COM には例外処理がありません。

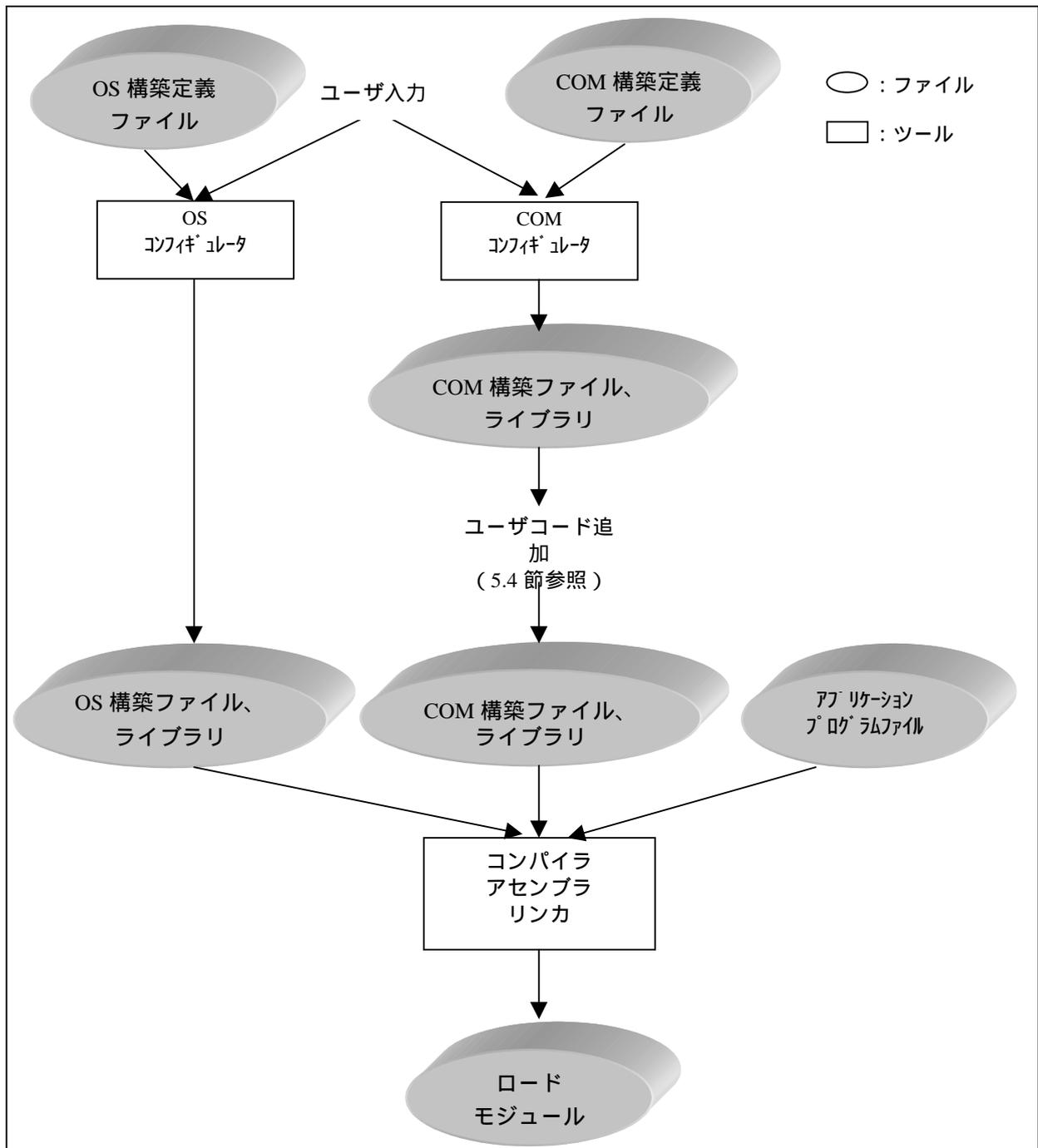
OS を使用しない場合、アプリケーションで例外を処理してください。

OS を使用する場合、OS が例外を処理します。詳細については、オペレーティングシステムマニュアルを参照してください。

4 システム構築

4.1 COMアプリケーションの構築

COM アプリケーションの構築手順を以下に示します。



コンフィギュレータは、ユーザ入力または構築定義ファイルをもとに構築ファイルおよびライブラリファイルを生成します。これらのファイルはアプリケーションプログラムファイルと共にコンパイル、アセンブル、結合され、ロードモジュールを生成します。

構築ファイルの生成方法については OS コンフィギュレータおよび COM コンフィギュレータのヘルプファイルを参照してください。OS 構築ファイルの詳細についてはオペレーティングシステムマニュアルを参照してください。

4.2 COM構築ファイル

COM 構築ファイルは次のファイルグループに分類されます。

- **App**
サンプルアプリケーションファイル
- **Com**
COM コンフィギュレータが生成した COM ライブラリ、COM オブジェクト、COM 構築ファイルおよびサンプルファイル
- **Os**
OS コンフィギュレータが生成したサンプル OS 構築ファイル

Com グループはさらに分類されます。

- **Evb**
サンプルターゲット HW 設定ファイル
- **Hcan**
HCAN ドライバファイル
- **Mcs**
サンプル CPU 初期化、セクション初期化ファイル
- **Msg**
メッセージパッケージファイル
- **Net**
ネットパッケージファイル

必要に応じて以下のファイルを書き換えてください。これらのファイルは COM コンフィギュレータで COM 構築ファイルを生成するたびに上書きされますのでご注意ください。

4.2.1 Evb

表 4.1 Evb グループ

ファイル	説明
int_hndl.c	割り込みハンドラ
ma_cpu.c	ハードウェア初期化
ma_cpu.h	ハードウェア初期化ヘッダ
ma_int.c	割り込み初期化
ma_int.h	割り込み初期化ヘッダ
ma_io.c	ポート設定
ma_io.h	ポート設定ヘッダ
ma_pfc.c	ピンファンクションコントローラ(PFC)設定
ma_pfc.h	ピンファンクションコントローラ(PFC)設定ヘッダ
makeapp.h	マクロ

4.2.2 Hcan

表 4.2 Hcan グループ

ファイル	説明
ma_hcan1.c	HCAN チャンネル 0 処理
ma_hcan1.h	HCAN チャンネル 0 ヘッダ
ma_hcan2.c	HCAN チャンネル 1 処理
ma_hcan2.h	HCAN チャンネル 1 ヘッダ
mac.h	HCAN マクロ
macan1.h	HCAN チャンネル 0 マクロ
macan2.h	HCAN チャンネル 1 マクロ

4.2.3 Mcs

表 4.3 Mcs グループ

ファイル	説明
dbsetc.src	セクション初期化定義
hwsetup.src	ハードウェア初期化
initsct.c	セクション初期化
intprg.src	共通割り込みプログラム
resetprg.src	リセット処理
stacksetc.src	リセットスタック定義
vect.inc	ベクタテーブルインクルードファイル
vecttbl.src	ベクタテーブル定義

4.2.4 Msg

表 4.4 Msg グループ

ファイル	説明
occallb.c	メッセージ初期化、混合送信用呼び出し関数
ocos.h	COM 割り込みレベル設定

5 プログラミング

以下に COM アプリケーションのプログラミング方法を示します。

5.1 COM初期化

COM の初期化は、以下に示された順序に従ってサービスおよび関数を呼び出してください。COM の初期化が終了するまでは COM 割り込みレベルを保つようにしてください。OS を使用しない場合、StartOS システムサービスの呼び出しは必要ありません。

```
/*--- HW Init ---*/
MA_Init_PFC();           /* ピンファンクションコントローラ(PFC)初期化 */
MA_Init_CPU();          /* ハードウェア初期化 */
MA_Init_IO();           /* ポート初期化 */

/*--- COM Init ---*/
StartCOM();             /* COM 初期化 */
MA_Init_INT();         /* 割り込み初期化 */

/*--- OS Init ---*/
StartOS(アプリケーションモード); /* OS 初期化 */
```

5.2 インクルードファイル

メッセージを使用するプログラムは必ず `ocmsg.h` ファイルをインクルードしてください。

5.3 割り込み

5.3.1 HCAN 割り込み

各 HCAN には、次の通り 4 つのベクタがあり、各割り込みがグループ分けされています。ベクタテーブルにそれぞれの ISR を登録してください。CCC0 で OS を使用する場合、カテゴリ 1 割り込みとして登録してください。CCC1 の場合、カテゴリ 2 割り込みとして登録してください。CCC1 では OS の割り込み前処理を介すので、ISR 名の最後に”_ISR”を付加して下さい。表 5.1 に HCAN ISR 名を示します。

- **ERS ベクタ**
 - エラーパッシブ割り込み
 - バスオフ割り込み
- **OVR ベクタ**
 - リセット割り込み
 - リモートフレーム要求割り込み
 - バス送信オーバーロードワーニング割り込み
 - バス受信オーバーロードワーニング割り込み
 - オーバーロードフレーム割り込み
 - 未読メッセージ割り込み
 - バス動作要求割り込み
- **RM ベクタ**
 - 受信メッセージ割り込み
- **SLE ベクタ**
 - 送信メールボックス空き割り込み

表 5.1 HCAN ISR 名

ベクタ	CCC0	CCC1
ERS	MA_IntHandler_ERS_HCAN x	MA_IntHandler_ERS_HCAN x _ISR
OVR	MA_IntHandler_OVR_HCAN x	MA_IntHandler_OVR_HCAN x _ISR
RM	MA_IntHandler_RM_HCAN x	MA_IntHandler_RM_HCAN x _ISR
SLE	MA_IntHandler_SLE_HCAN x	MA_IntHandler_SLE_HCAN x _ISR

【注】 x : 1 (HCAN チャンネル 0 用)または 2 (HCAN チャンネル 1 用)

5.3.2 割り込み優先レベル

- COM 割り込みレベル

通信サービスを使用するプログラムの中で最高の割り込み優先レベルを COM 割り込みレベルに設定してください。CCC0 で OS を使用する場合、OS 割り込みレベル以上に設定してください。CCC1 の場合、OS 割り込みレベルが COM 割り込みレベルに設定されます。COM 割り込みレベルより高いプログラムから通信サービスを発行しないでください。COM 割り込みレベルの設定方法については Msg ディレクトリの ocos.h を参照してください。

- HCAN 割り込みレベル

HCAN のチャンネル 0 とチャンネル 1 の割り込み優先レベルは同じにして下さい。HCAN 割り込みレベルは COM 割り込みレベルと同じにして下さい。設定方法についてはサンプル提供の ma_int.c を参照してください。

5.4 ユーザコード

COM 構築ファイルは、COM コンフィギュレータによって生成されますが、以下の処理はユーザコードを追加することが可能です。

- 初期化処理
- コールバック関数
- COM で予約されていない HCAN 割り込み

5.4.1 初期化処理

MA_Init_PFC()

ma_pfc.c ファイルに存在します。サンプル提供の main.c から呼び出しています。

MA_Init_CPU()

ma_cpu.c ファイルに存在します。サンプル提供の main.c から呼び出しています。

MA_Init_IO()

ma_io.c ファイルに存在します。サンプル提供の main.c から呼び出しています。

MA_Init_INT()

ma_int.c ファイルに存在します。サンプル提供の main.c から呼び出しています。

5.4.2 コールバック関数

MessageInit()

occallb.c ファイルに存在します。この関数は、StartCOM サービスから呼び出されます。詳細は6.1.2節を参照してください。

OC_MixedTxEval()

occallb.c ファイルに存在します。この関数は、混合送信モードでメッセージを送信時、COM 内から呼び出さ

れます。詳細は6.2.2節を参照してください。

5.4.3 HCAN ISR

以下の HCAN 割り込みを処理するためにコードを追加することができます。割り込み要因の詳細は、ハードウェアマニュアルを参照してください。

表 5.2 HCAN 割り込み用ユーザコード

割り込み要因	ユーザコード追加場所
未読メール	int_hndl.c: MY_UnReadMailHandler1() (HCAN チャンネル 0 用) MY_UnReadMailHandler2() (HCAN チャンネル 1 用)
リセット	int_hndl.c: MY_PowerUpHandler1() (HCAN チャンネル 0 用) MY_PowerUpHandler2() (HCAN チャンネル 1 用)
オーバーロードフレーム	macan1.h: MA_OVERLOAD_FRAME_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_OVERLOAD_FRAME_USERCODE2 マクロ (HCAN チャンネル 1 用)
バスオフ	macan1.h: MA_BUS_OFF_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_BUS_OFF_USERCODE2 マクロ (HCAN チャンネル 1 用)
エラーパッシブ	macan1.h: MA_ERROR_PASSIVE_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_ERROR_PASSIVE_USERCODE2 マクロ (HCAN チャンネル 1 用)
バス受信オーバーロードワーニング	macan1.h: MA_REC_WARNING_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_REC_WARNING_USERCODE2 マクロ (HCAN チャンネル 1 用)
バス送信オーバーロードワーニング	macan1.h: MA_TEC_WARNING_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_TEC_WARNING_USERCODE2 マクロ (HCAN チャンネル 1 用)
バス動作要求	macan1.h: MA_WU_BUS_ACTIF_USERCODE1 マクロ (HCAN チャンネル 0 用) macan2.h: MA_WU_BUS_ACTIF_USERCODE2 マクロ (HCAN チャンネル 1 用)

5.5 OSとのインタフェース

5.5.1 割り込み

次のカテゴリ規則にしたがって、COM (通信サービス) を使用する割り込みを OS コンフィギュレータに登録してください。

表 5.3 カテゴリ規則

割り込み	CCC0	CCC1
HCAN 割り込み	カテゴリ 1	カテゴリ 2
その他の割り込み	カテゴリ 1 または 2	カテゴリ 1 または 2

5.5.2 アラームとタスク

周期および混合送信モード用の周期アラームおよびタスクはユーザアプリケーションで用意してください。そのタスクを OS コンフィギュレータに登録してください。

5.6 レジスタ

通信サービスは R0 ~ R7, FR0 ~ FR11, FPUL, FPSCR¹ レジスタを保証しません。これらのレジスタを通信サービス呼び出し後に使用する場合は、あらかじめ保存しておいてください。

5.7 スタック

COM はスタックを変更しません。したがって、通信サービスを呼び出すプログラムのスタックに各サービスのスタックサイズを加算してください。また、各割り込みのスタックサイズを割り込みスタックに加算してください。スタックサイズの詳細については Ho7000 シリーズリリースノートを参照してください。

¹ FR0 ~ FR11, FPUL, FPSCR レジスタは浮動小数点演算ユニット (FPU) を搭載しているプロセッサのみ有効です。

5.8 アセンブラルーチンからの通信サービスコール

アセンブラルーチンから通信サービスを呼び出す場合、JSR 命令によって各通信サービスの先頭アドレスに分岐してください。

その際、以下のパラメータルールに従ってください。パラメータの型については、表 7.4 データ型および構築ファイルの Msg グループの C 言語ヘッダファイルを参照してください。

表 5.4 引数

レジスタ	引数の数
R4	第 1 引数
R5	第 2 引数
R6	第 3 引数
R7	第 4 引数

戻り値は R0 レジスタに格納されます。

通信サービス呼び出し前後では R0 ~ R7, PR, FR0 ~ FR11, FPUL, FPSCR レジスタは保証されません。したがって、これらのレジスタを使用する場合、通信サービス呼び出し前に保存し、呼び出し後に復帰してください。

5.9 アセンブラ使用時の注意事項

アセンブラルーチンから COM を初期化する場合、MA_Init_PFC()関数などの各初期化処理を JSR で呼び出してください。また、構築ファイルの Evb グループ、Mcs グループをアセンブリ言語で記述することはできますが、Hcan グループ、Msg グループ、Net グループは必ず C 言語で記述してください。

6 API

以下に通信サービスのインタフェースを示します。アセンブラインタフェースについては5.8節を参照してください。

6.1 標準インタフェース

6.1.1 StartCOM

構文： `StatusType StartCOM(void)`

パラメータ（入力）：

なし

パラメータ（出力）：

なし

説明：

COMを初期化します。本サービスは、最後にコールバック関数 `MessageInit()` を呼び出します。

エラーステータス：

標準

正常終了 `E_OK`

拡張（拡張ステータスで追加されるコード）

なし

6.1.2 MessageInit

構文： StatusType MessageInit(void)

パラメータ（入力）：

なし

パラメータ（出力）：

なし

説明：

この関数は StartCOM サービスによって COM が自動的に呼び出します。本関数は occallb.c ファイルによって提供されています。必要に応じて書き換えてください。

エラーステータス：

標準

正常終了 E_OK

初期化失敗 E_COM_SYS_MSG_INIT_FAILED

拡張（拡張ステータスで追加されるコード）

なし

6.1.3 SendMessage

構文： StatusType SendMessage(OC_SymbolicNameT <Msg>, OC_DataRefT <Data>)

パラメータ（入力）：

Msg メッセージ名称

Data 送信データが格納された領域のアドレス

パラメータ（出力）：

なし

説明：

送信データはアプリケーションで指定したデータ領域から、COM 内のメッセージデータ領域へコピーされます。ネットワークメッセージの場合、ネットワークへ送信を開始します。この場合、送信完了前に本サービスは復帰します。周期送信または混合送信で使用可能な周期メッセージを送信した場合、ネットワークへの送信を開始しません。ネットワークへの送信開始は、周期タスクから OC_SendMsgToNetPer 通信サービスを呼び出すことにより行われます。また、混合送信の場合、ネットワークへの送信を条件付で開始することが可能です。

エラーステータス：

標準

正常終了	E_OK
メッセージがロックされています	E_COM_LOCKED
通知としてイベントをセットすることができません	E_COM_SYS_EVENT_SETTING_DENIED
通知としてタスクを起動することができません	E_COM_SYS_TASK_ACTIVATION_DENIED
タイムアウト監視のアラームを開始できません	E_COM_SYS_ALARM_START_DENIED
ネットパッケージによって送信が拒否されました	E_COM_SYS_NET_TX_DENIED

拡張（拡張ステータスで追加されるコード）

無効メッセージ	E_COM_ID
---------	----------

6.1.4 ReceiveMessage

構文： StatusType ReceiveMessage(OC_SymbolicNameT <Msg>, OC_DataRefT <Data>)

パラメータ（入力）：

Msg メッセージ名称

Data 受信データ領域およびそのアドレス

パラメータ（出力）：

*Data 受信したデータ

説明：

受信データが COM 内のメッセージデータ領域から、アプリケーションで指定したデータ領域へコピーされます。このサービスがメッセージデータの受信を待つことはありません。COM 内に存在する最後に受信されたメッセージデータを単にコピーします。

エラーステータス：

標準

正常終了

E_OK

メッセージデータは、初期化された後、一度も更新されていない

E_COM_NOMSG

拡張（拡張ステータスで追加されるコード）

無効メッセージ

E_COM_ID

6.1.5 GetMessageStatus

構文： StatusType GetMessageStatus(OC_SymbolicNameT <Msg>)

パラメータ（入力）：

Msg メッセージ名称

パラメータ（出力）：

なし

説明：

メッセージの状態を取得します。

エラーステータス：

標準

正常終了

E_OK

メッセージがロックされています

E_COM_LOCKED

メッセージデータは、初期化された後、一度も更新されていない

E_COM_NOMSG

拡張（拡張ステータスで追加されるコード）

無効メッセージ

E_COM_ID

6.2 独自インタフェース

OSEK COM 仕様で定義されている通信サービスに加え、独自のインタフェースを提供しています。

6.2.1 OC_SendMsgToNetPer

構文：

```
enum OC_ILOpResultE OC_SendMsgToNetPer(OC_SymbolicNameT <Msg>)
```

パラメータ（入力）：

Msg メッセージ名称

パラメータ（出力）：

なし

説明：

周期メッセージのネットワークへの送信を開始します。周期または混合送信用メッセージのみ使用できます。周期的に起動されるタスクから本サービス呼び出ししてください。本サービスによってネットへ周期的に送信する前に必ず SendMessage 通信サービスを発行してください。SendMessage 通信サービスによって送信されたメッセージデータが周期的に送信されます。SendMessage 通信サービスを発行しない場合、本サービスはエラーを返さず不定データをネットへ送信します。

エラーステータス：

標準

正常終了

OC_ILOpResOK

メッセージがロックされています

OC_ILOpResMsgDsInUse

タイムアウト監視のアラームを開始できません

OC_ILOpResError

ネットパッケージによって送信が拒否されました

OC_ILOpResNetTxDenied

拡張（拡張ステータスで追加されるコード）

無効メッセージ

OC_ILOpResError

6.2.2 OC_MixedTxEval

構文：

```
OC_CallbStatusT OC_MixedTxEval(OC_SymbolicNameT <pMsgH>)
```

パラメータ（入力）：

pMsgH メッセージ名称

パラメータ（出力）：

なし

説明：

ネットワークへの送信開始の可否を COM に指示するための関数です。混合送信用メッセージのみ使用できません。この関数は SendMessage 通信サービスによるメッセージ送信時、COM 内から自動的に呼び出されます。本関数は occallb.c ファイルによって提供されていますので、送信条件を評価するコードを追加してください。pMsgH は整数ではないため switch 文での判定はできません。if 文で判定してください。評価の結果は、次のリターン値のいずれかになるようにしてください。

エラーステータス：

標準

ネットへの送信を開始する

OC_TxCondIsSendD

ネットへの送信を開始しない

OC_TxCondIsNoSendD

拡張（拡張ステータスで追加されるコード）

なし

7 付録

7.1 通信サービスリターンコード

表 7.1 通信サービスリターンコード

通信サービス	標準エラーステータス	拡張エラーステータスで追加されるコード
StartCOM	E_OK	---
MessageInit	E_OK E_COM_SYS_MSG_INIT_FAILED*	---
SendMessage	E_OK E_COM_LOCKED E_COM_SYS_EVENT_SETTING_DENIED* E_COM_SYS_TASK_ACTIVATION_DENIED* E_COM_SYS_ALARM_START_DENIED* E_COM_SYS_NET_TX_DENIED*	E_COM_ID
ReceiveMessage	E_OK E_COM_NOMSG	E_COM_ID
GetMessageStatus	E_OK E_COM_NOMSG E_COM_LOCKED	E_COM_ID
OC_SendMsgToNetPer*	OC_ILOpResOK* OC_ILOpResMsgDsInUse* OC_ILOpResError* OC_ILOpResNetTxDenied*	OC_ILOpResError*
OC_MixedTxEval*	OC_TxCondIsSendD* OC_TxCondIsNoSendD*	---

【注】* 独自機能

7.2 リターンコードID

表 7.2 リターンコード ID

リターンコード	説明	ID
E_OK	正常終了	0 (H'0)
E_COM_ID	無効メッセージ	1 (H'1)
E_COM_LIMIT	将来用 (未使用)	2 (H'2)
E_COM_NOMSG	メッセージデータは、初期化された後、一度も更新されていない	3 (H'3)
E_COM_LOCKED	メッセージがロックされています	4 (H'4)
E_COM_SYS_NET_TX_DENIED*	ネットパッケージによって送信が拒否されました	64 (H'40)
E_COM_SYS_EVENT_SETTING_DENIED*	通知としてイベントをセットすることができません	65 (H'41)
E_COM_SYS_TASK_ACTIVATION_DENIED*	通知としてタスクを起動することができません	66 (H'42)
E_COM_SYS_ALARM_START_DENIED*	タイムアウト監視のアラームを開始できません	67 (H'43)
E_COM_SYS_UNEXPECTED_STATE*	将来用 (未使用)	68 (H'44)
E_COM_SYS_MSG_INIT_FAILED*	初期化失敗	69 (H'45)
OC_ILOpResOK*	正常終了	256(H'100)
OC_ILOpResError*	無効メッセージ タイムアウト監視のアラームを開始できません	257 (H'101)
OC_ILOpResMsgDsInUse*	メッセージがロックされています	258 (H'102)
OC_ILOpResNetTxDenied*	ネットパッケージによって送信が拒否されました	259 (H'103)
OC_ILOpResMsgUnexpectedState*	将来用 (未使用)	260 (H'104)
OC_ILOpResToNetMessageNotSupported*	将来用 (未使用)	261 (H'105)
OC_ILOpResFrNetMessageNotSupported*	将来用 (未使用)	262 (H'106)
OC_TxCondIsSendD*	ネットへの送信を開始する	1 (H'1)
OC_TxCondIsNoSendD*	ネットへの送信を開始しない	2 (H'2)

【注】* 独自機能

7.3 通信サービス呼び出し

呼び出しプログラムにおいて発行可能な通信サービスを で示します。”---”は動作を保証しないことを意味します。

表 7.3 通信サービス呼び出し

通信サービス	COM 初期化前 (必須) CCC0/CCC1	CCC0	CCC1						
			タ ス ク	ISR	Error フ ック	PreTask フ ック	PostTask フ ック	Startup フ ック	Shutdown フ ック
StartCOM		---	---	---	---	---	---	---	---
SendMessage	---				---	---	---	---	---
ReceiveMessage	---					---	---	---	---
GetMessageStatus	---					---	---	---	---
OC_SendMsgToNetPer*	---	---		---	---	---	---	---	---

【注】* 独自機能

7.4 データ型

表 7.4 データ型

名称	型 (サイズ)	説明
StatusType	unsigned long (4 バイト)	リターン値
OC_SymbolicNameT	struct OC_MsgHandleS * (4 バイト)	メッセージ名称
OC_DataRefT	void * (4 バイト)	メッセージデータのアドレスまたはポインタ
OC_ILOpResultE	enum (4 バイト)	リターン値
OC_CallbStatusT	unsigned char (1 バイト)	リターン値

7.5 制限値

表 7.5 制限値

項目	制限
COM 当たり最大ネット数	2
最大 Local メッセージ数	1000
ネット当たり最大 To Net メッセージ数	15
ネット当たり最大 From Net メッセージ数	31
メールボックス 0 用最大メッセージ数	16
メールボックス 1~15 用各最大メッセージ数	1
ネット当たり最大メッセージ数	31

Ho7000 シリーズ 通信マニュアル



ルネサスエレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668