

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ひ申し上げます。

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、誤りが無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

# H8S/2612F 内蔵CANによるフラッシュメモリの書換え アプリケーションノート

## はじめに

本アプリケーションノートでは、ユーザプログラムモードを使用してフラッシュメモリを書き換える方法を説明します。書き込みデータの供給は、H8Sシリーズ内蔵HCAN（日立コントローラエリアネットワーク）を使用した場合の動作例を説明します。

フラッシュメモリ、HCANの詳細については、「H8S / 2612F ハードウェアマニュアル」の各章を合わせてご参照ください。

- ・ROM
- ・HCAN

本アプリケーションノートに掲載されているプログラム、回路例等の動作は確認しておりますが、実際にご使用になる場合は、改めて動作確認のうえ、ご使用くださいますようお願いいたします。（ただし、本文中のプログラム例は、H8S / 2612Fの内蔵HCANについて記述しています。）

Preliminary

<b>1 . 概要</b> .....	<b>1</b>
1 . 1 HCAN ユニットを内蔵した F-ZTAT マイコン ( H8S シリーズ ) の一覧 .....	1
1 . 2 ユーザプログラムモードの概要 .....	2
1 . 3 ユーザプログラムモードでの書き換え方式の概要 .....	3
<b>2 . サンプルシステムの概要</b> .....	<b>4</b>
2 . 1 ハードウェア一覧 .....	4
2 . 2 ソフトウェア一覧 .....	5
2 . 3 サンプルプログラムのカスタマイズ項目 .....	5
2 . 4 このアプリケーションノートで使用するプログラムのインストール .....	6
<b>3 . ユーザプログラムモードでのフラッシュメモリの書き換え方式</b> .....	<b>7</b>
3 . 1 アプリケーション ( サンプル ) プログラムのフラッシュメモリのマップ .....	8
3 . 2 ターゲットボード、SCI HCAN 通信変換ボードへのプログラム書き込み .....	8
3 . 3 このアプリケーションノートでのCAN通信設定 .....	9
3 . 4 ユーザプログラムモードでのフラッシュメモリの書き換えシーケンス .....	10
<b>4 . ユーザプログラムモードでのフラッシュメモリ書き換え動作の説明</b> .....	<b>12</b>
4 . 1 初期状態 .....	12
4 . 2 書き込み / 消去制御プログラムの転送 .....	13
4 . 3 フラッシュメモリのブロック消去 .....	14
4 . 4 新アプリケーションプログラムの書き込み .....	15
<b>5 . ソフトウェアの詳細説明</b> .....	<b>16</b>
5 . 1 F-ZTAT MICROCOMPUTER H-CAN PROGRAM .....	16
5 . 2 SCI HCAN 通信変換プログラム .....	17
5 . 3 アプリケーション ( サンプル ) プログラム .....	18
<b>6 . アプリケーション ( サンプル ) プログラムの作成例</b> .....	<b>19</b>
6 . 1 関数、変数一覧 .....	19
6 . 2 CAN 通信設定の変更例 .....	19
6 . 3 CAN 通信の受信 / 送信メールアドレス番号の変更例 .....	21
6 . 4 アプリケーション ( サンプル ) プログラムフローチャート .....	25
<b>7 . 書き込み / 消去制御プログラムの作成例</b> .....	<b>26</b>
7 . 1 消去 ( Eビット )、書き込み ( Pビット ) の印加時間、SWEビットのウェイト時間等の計算 .....	26
7 . 2 関数、変数及び、定数一覧 .....	29
7 . 3 CAN 通信の受信 / 送信メールアドレス番号の変更例 .....	31
7 . 4 書き込み / 消去プログラムフローチャート .....	35

8 . オンボード書き込みツール及び、SCI HCAN 通信変換プログラムの機能及び、操作説明 .....	38
8 . 1 オンボード書き込みツールのインストール .....	38
8 . 2 ターゲットボードへのアプリケーション ( サンプル ) プログラムの初回書き込み .....	42
8 . 3 SCI HCAN 通信変換ボードへの SCI HCAN 通信変換プログラムの書き込み .....	48
8 . 4 ユーザプログラムモードでのフラッシュメモリ書き換え .....	50
8 . 5 FLASHCAN.EXE のエラーメッセージ ( HCAN 対応の追加分 ) .....	62
9 . 補足説明 .....	63
9 . 1 ユーザプログラムモードでフラッシュメモリを書き換えるために必要な項目 .....	63
9 . 2 ユーザプログラムモードとブートモードとの違い .....	64
9 . 3 E ビット、P ビットの印加時間の実測方法 .....	65

Preliminary

## 1. 概要

このアプリケーションノートでは、H8S/2612F の内蔵フラッシュメモリをユーザプログラムモードで書き換える方法について説明します。書き込みデータは H8S/2612F の内蔵 HCAN を使用して供給します。

サンプルシステムとして、図 1 に示すシステム構成でフラッシュメモリを書き換える方法を説明します。また、ユーザシステムの動作周波数や CAN バス仕様に合わせて、このアプリケーションノートに記載しているプログラムをカスタマイズする方法を説明します。

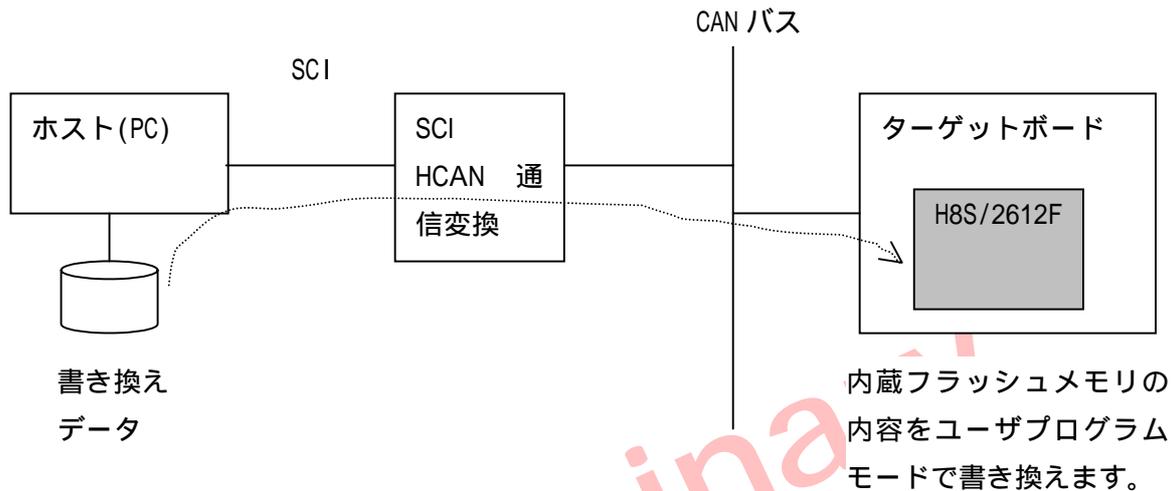


図 1 サンプルシステムの構成図

### 1.1 HCAN ユニートを内蔵した F-ZTAT マイコン (H8S シリーズ) の一覧

このアプリケーションノートは、次のデバイスに応用できます。

- H8S/2612F
- H8S/2623F
- H8S/2626F
- H8S/2636F

H8S/2612F 以外のマイコンを利用するときの参考として、このアプリケーションノートを使用するときには、次の相違個所についてご注意ください。

- 内蔵レジスタのアドレス、ビット位置、機能の相違
- フラッシュメモリの消去 / 書き込み制御方式 (E ビット、P ビットの印加時間、等) の相違

## 1.2 ユーザプログラムモードの概要

ユーザプログラムモードは、オンボードでフラッシュメモリを書き換えるモードです。ユーザボードにF-ZTATマイコンを実装した状態で、内蔵フラッシュメモリの内容を書き換えることができます。

オンボードでフラッシュメモリを書き換えるモードは、ブートモードとユーザプログラムモードの2通りがあります。ブートモードでは、F-ZTATマイコンに内蔵されているブートプログラムが実行されフラッシュメモリの書き込みを実現します。それに対し、ユーザプログラムモードでは、フラッシュメモリ(内蔵ROM)上のアプリケーションプログラムが実行されます。このため、あらかじめアプリケーションプログラムに書き換え処理を組み込んでおく必要があります。

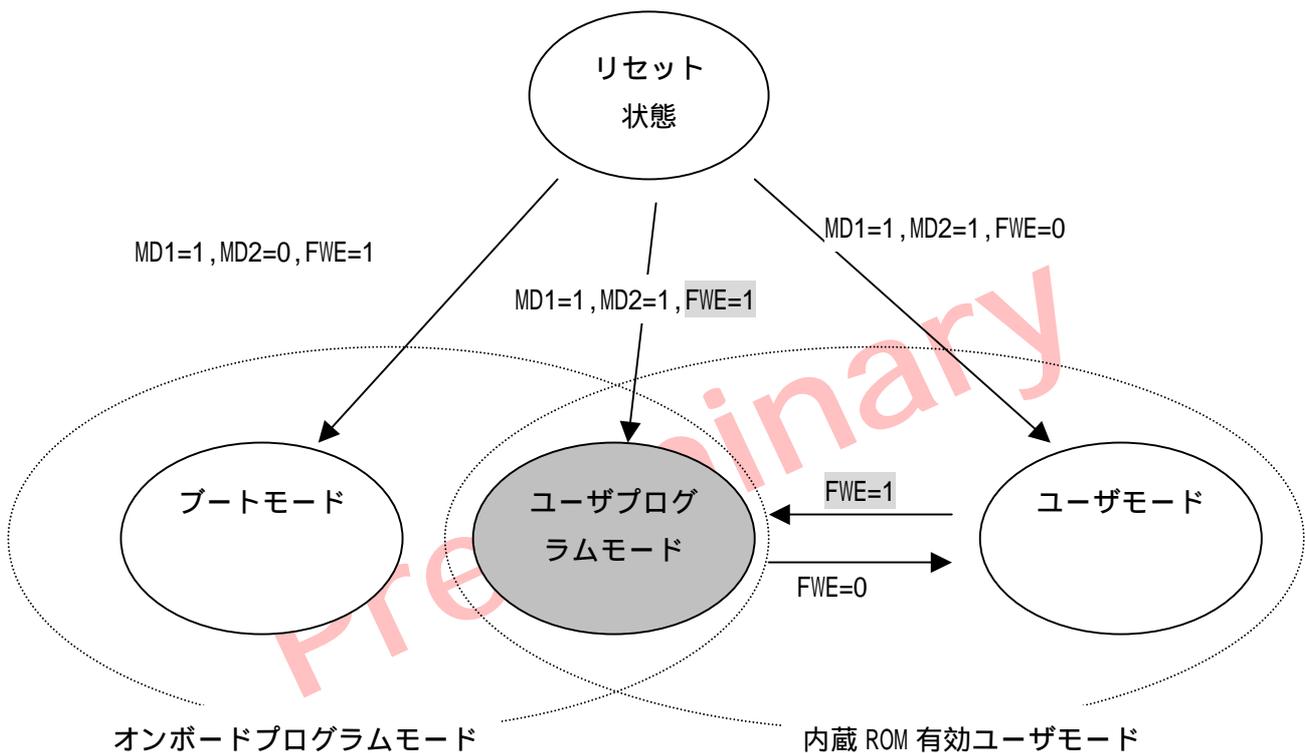


図 2 ユーザプログラムモードの遷移図

ブートモードでは、フラッシュメモリの書き込み前に内蔵のブートプログラムによってフラッシュメモリの全面が自動的に消去されます。このためアプリケーションプログラムの一部分を書き換える場合でも全面的に書き換える必要があります。

ユーザプログラムモードでは、ユーザシステムに合わせて消去 / 書き込みを任意に実行することができるので、消去ブロック単位での部分的な書き換えが可能です。

### 備考

ユーザプログラムモードの"プログラム"とはフラッシュメモリの"書き込み"の意味です。

### 1.3 ユーザプログラムモードでの書き換え方式の概要

ユーザプログラムモードでは内蔵ROMが有効となるので、書き換え処理を組み込んだアプリケーションプログラムをあらかじめ内蔵フラッシュメモリに書き込んでおく必要があります。この初回の書き込みは、ブートモードまたは、ROMライターモードで書き込みます。また誤操作などにより、書き換え処理を組み込んだアプリケーションプログラムを消去してしまいユーザプログラムモードでの書き換えができなくなった場合など、ブートモードによって強制的に書き込みが可能です。

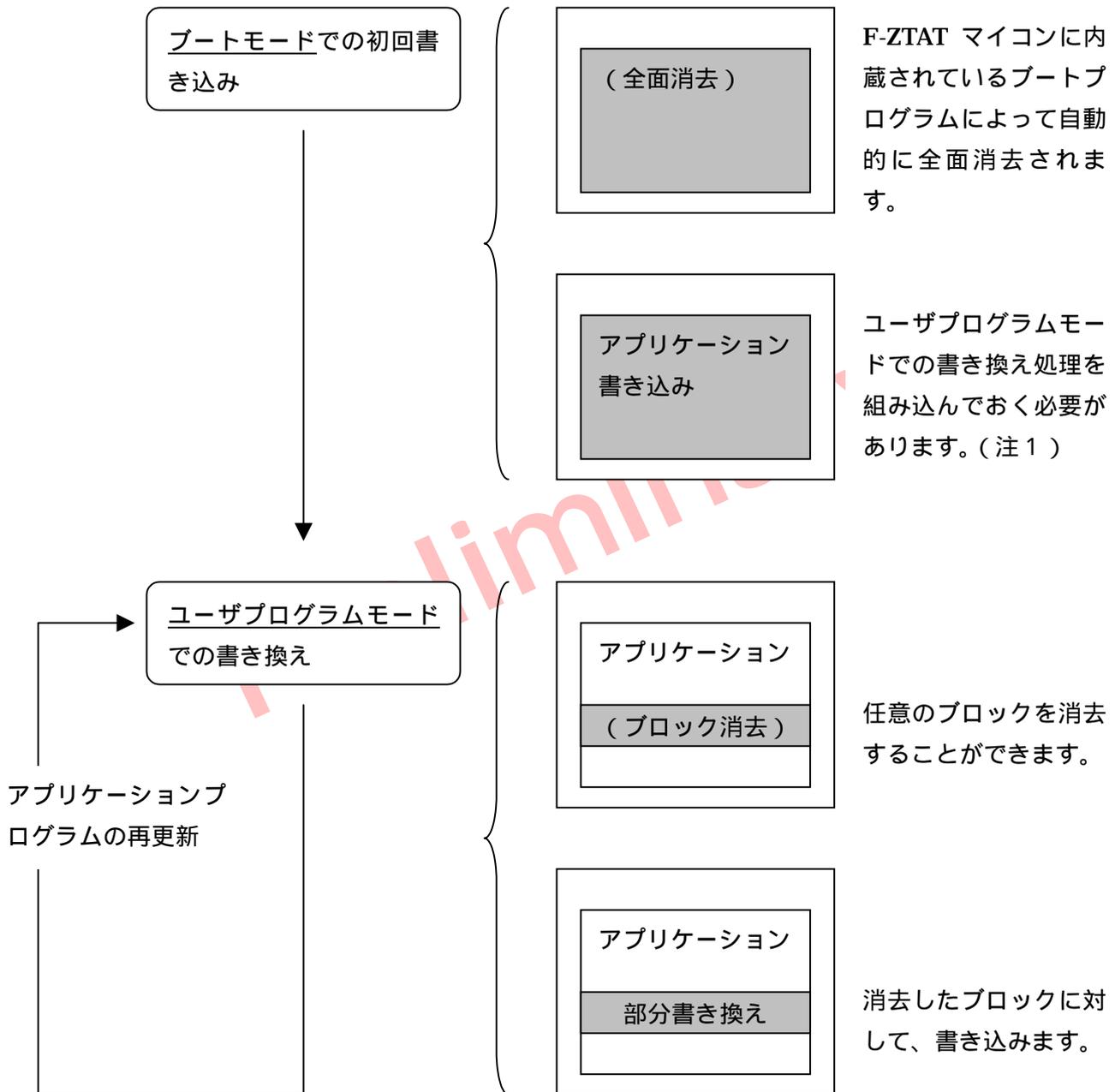


図 3 ユーザプログラムモードでの書き込み方式 (概要)

(注1) アプリケーションへ組み込んでおく書き換え処理の内容

書き換え処理の遷移条件の検出及び、「書き込み/消去制御プログラム」をRAMへ転送し、RAMへジャンプする処理をアプリケーションへ組み込んでおく必要があります。

## 2. サンプルシステムの概要

### 2.1 ハードウェア一覧

このアプリケーションノートでのサンプルプログラムを実行するためには、次の構成が必要不可欠です。

**表 1 ハードウェア一覧**

	ハードウェア物品名	仕様	備考
1	ホスト ( P C )	FlashCAN.exe を実行します。 シリアルインタフェースを実装している ことが必要です。	DOS/V パソコン OS : Windows 98(S), 2000, NT4.0, ME, XP
2	SCI CAN 通信変換ボード	ホストからのシリアル通信を CAN 通信に 変換して、ターゲットボードと通信しま す。 また、ターゲットボードからの CAN 通信 をシリアル通信に変換します。	(株)北斗電子社製 「LIN・CAN スタータキット (H8S/2612F)」
3	ターゲットボード	書き込み先のフラッシュメモリを内蔵し た H8S/2612F をオンボード実装していま す。	(株)北斗電子社製 「LIN・CAN スタータキット (H8S/2612F)」  ユーザプログラムモードに対応す るために FWE 端子の切替えスイッ チの追加が必要です。(*1)
4	シリアルケーブル	9 ピン ホストと SCI CAN 通信変換ボード J3 コ ネクタを接続。	(株)北斗電子社製 「LIN・CAN スタータキット (H8S/2612F)」に付属
5	CAN バスケーブル	3 ピン SCI CAN 通信変換ボード J7 とターゲッ トボード J7 を接続。	(株)北斗電子社製 「LIN・CAN スタータキット (H8S/2612F)」に付属

(\*1)

ユーザプログラムモードへの遷移の為に FWE 端子切替えは、(株)北斗電子社製の「LIN・CAN スタータキット(H8S/2612F)」では、ボード上スライドスイッチ (SW11) で行います。

その他、モード変更についての具体的な操作は製品付属の取扱説明書にてご確認ください。「LIN・CAN スタータキット(H8S/2612F)」の回路図は、ボードに付属されているものでご確認ください。

## 2.2 ソフトウェア一覧

**表 2 ソフトウェア一覧**

	ファイル名	プログラム名称	備考
1	FlashCAN.exe	F-ZTAT Microcomputer H-CAN Program	ホスト（PC）で実行します。 このアプリケーションノートで使用する ための評価版です。
2	SCI2612F3.s ub	書き込み制御プログラム （SCI 通信）	ブートモードでの初回の書き込みに使用 します。 FlashCAN.exe からシリアル送信し、ター ゲットボードのブートプログラムにより 受信 RAM 格納し実行します。
3	HCAN2612F3.s ub (*1)	書き込み / 消去制御プログラム （HCAN 通信）	ユーザプログラムモードでの書き換えで 使用します。 FlashCAN.exe からシリアル送信し、ター ゲットボードのアプリケーションに組み 込まれた処理により受信 RAM 格納し実 行します。
4	SCItoCAN.mot	SCI CAN 通信変換プログラム	SCI CAN 通信変換ボードで実行します。
5	Sample1.mot (*1)	アプリケーション（サンプル）プログラム	ターゲットボードで実行します。 ユーザプログラムモードでの書き換え処 理が組み込まれています。

(\*1) このアプリケーションノートのサンプルプログラムをユーザシステムに合わせて使用するために、次の項目のカスタマイズが可能です。ソースファイルの変更及び、コンパイル、アセンブルしてください。サンプルシステムをそのまま使用する場合には、カスタマイズは不要です。

## 2.3 サンプルプログラムのカスタマイズ項目

	カスタマイズ項目	デフォルト設定	変更箇所
1	CAN の通信設定	（次ページ参照）	・ Sample1.mot HCAN_up.c の InitHCAN()関数
2	内蔵 HCAN で使用する 送受信のメールボッ クス番号	受信：MB4 送信：MB5	・ Sample1.mot HCAN_up.src の InitHCAN()、PowerON_Reset()、 CAN_MB4_rcv1byte()、CAN_MB5_trs1byte()関数 ・ HCAN2612f3.mot HCAN2612f3.src の RCV1BYTE()、TRS1BYTE() サブルーチン

(注1) コンパイル、アセンブルは、DOS プロンプトのバッチファイルか Hew プロジェクトで実行します。  
添付のバッチファイル / Hew プロジェクトは、コンパイラパッケージ Ver.5.0.02を使用した例です。  
コンパイラパッケージのバージョンに合わせて変更してください。

- ・ HCAN2612f3.mot の変更には、[HCAN2612f3]フォルダの HCAN2612f3.bat / Hew2.hws を使用します。
- ・ Sample1.mot の変更には、[Sample1]フォルダの Sample1.bat / Hew2.hws を使用します。

## 2.4 このアプリケーションノートで使用するプログラムのインストール

インストールは、setup.exe を実行してください。インストール後のディレクトリ構成を示します。

### 【標準インストール後のディレクトリ構成】

```
[c:¥FlashCAN]   FlashCAN.exe
                 SCI2612F3.sub
                 SCI2612F3.inf
                 HCAN2612F3.sub
                 HCAN2612F3.inf
                 [HCAN2612F3]           ; 書き込み / 消去制御プログラム
                 HCAN2612F3.src . . . (*1)カスタマイズ対象
                 HCAN2612F3.bat
                 |
                 [Hew2]                 [Debug]                 Debug.hdp
                 Hew2.hws                [Release]               Release.hdp
                 Hew2.tws                [SimDebug_H8S-2600A]    HCAN2612f3.abs
                 Hew2.hbp                Hew2.hwp                 HCAN2612f3.h8a
                                     Hew2.pgs                 HCAN2612f3.hlk
                                     Hew2.tps                 HCAN2612f3.obj
                                     DefaultSession.hsf      Hew2.map
                                     SimSessionH8S-2600A.hsf
                 [Sample1]              ; アプリケーション ( サンプル ) プログラム
                 Sample1.mot
                 HCAN_up.c . . . (*1)カスタマイズ対象
                 Sample1.c
                 io_2612.h
                 Sample1.bat
                 |
                 [Hew2]                 [Debug]                 Debug.hdp
                 Hew2.hws                [Release]               Release.hdp
                 Hew2.tws                [SimDebug_H8S-2600A]    Sample1.abs
                 Hew2.hbp                Hew2.hwp                 HCAN_up.h8c
                 Data_00.mot             Hew2.pgs                 Sample1.h8c
                 Data_FF.mot            Hew2.tps                 Sample1.hlk
                                     DefaultSession.hsf      HCAN_up.obj
                                     SimSessionH8S-2600A.hsf  Sample1.obj
                                     Hew2.map
                 [SCItoCAN]              ; SCI CAN 通信変換プログラム
                 SCItoCAN.mot
```

書き込みツール(FlashCAN.exe)は、ウィンドウズのスタートメニューから起動できます。

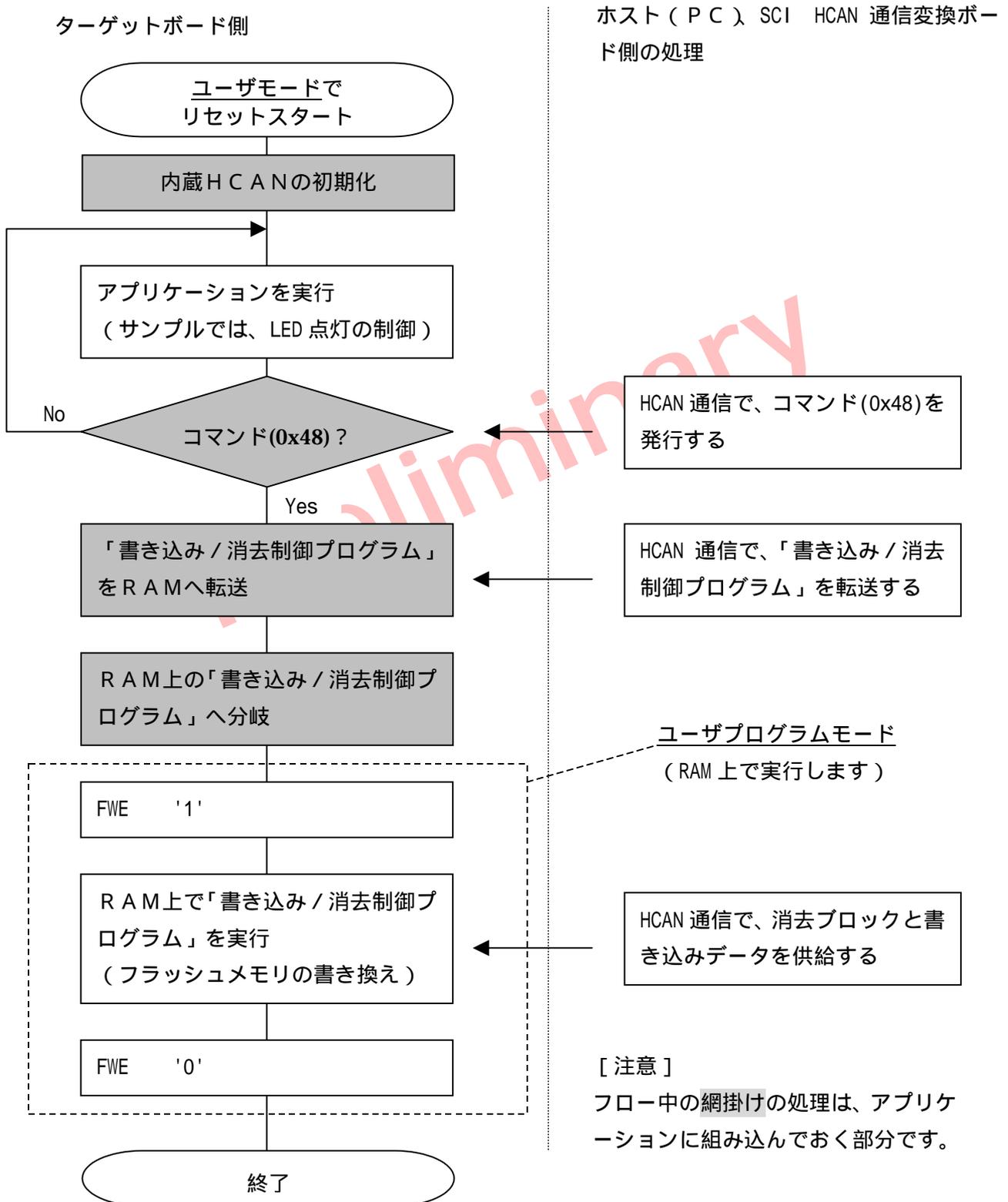
[ スタート ] メニュー      プログラム(P)      FlashCAN      FlashCAN

### 3. ユーザプログラムモードでのフラッシュメモリの書き換え方式

このアプリケーションノートでは、下記的手段でフラッシュメモリを書き換えます。

- HCAN通信でのコマンド受信をトリガとして、フラッシュメモリの書き換え処理の実行を開始する
- HCAN通信で外部から「書き込み/消去制御プログラム」をRAMへ転送する
- ターゲットボードのFWE端子をスイッチで切り換える (FWE='1'でユーザプログラムモードへ遷移)
- HCAN通信で外部から書き込みデータを転送する

また、、 のホストとして、ホスト (PC) 及び、SCI HCAN通信変換ボードを使用します。



### 3.1 アプリケーション（サンプル）プログラムのフラッシュメモリのマップ

H'000000	EB0(1kバイト) ベクタテーブル	} ベクタテーブル
H'0003FF H'000400	EB1(1kバイト) 未使用	
H'0007FF H'000800	EB2(1kバイト) アプリケーションの本体	} プログラムの本体：Sample1.mot (ブートモードで書き込み)
H'000BFF H'000C00	EB3(1kバイト) サンプルプログラム	
H'000FFF H'001000	EB4(28kバイト) パラメータデータ	} パラメータデータ：Data_FF.mot/Data_00.mot (ユーザプログラムモードで書き換え) このアプリケーションノートでの書き換える 対象はこのブロックとします。
H'007FFF H'008000	EB5 ： 未使用	
H'01FFFF	EB9	

#### 備考

Sample1.mot は、ブートモードでターゲットボードのフラッシュメモリへダウンロードしておきます。  
このアプリケーションノートでは、ユーザプログラムモードで Data\_FF.mot または、Data\_00.mot をフラッシュメモリへ書き換えます。アプリケーションはこのデータを参照して、LED の点灯 / 点滅を切り換えますので、書き換わったことを目視確認できます。

### 3.2 ターゲットボード、SCI HCAN 通信変換ボードへのプログラム書き込み

ユーザプログラムモードでの書き換えを行なう前に、ブートモードで各プログラムを書き込んでおく必要があります。ブートモードでの書き込みは、FlashCAN.exeの「Standard Mode」を使用します。

#### [ブートモードでの書き込み]

- ・Sample1.mot     ターゲットボードへ書き込む
- ・SCItoCAN.mot    SCI HCAN通信変換ボードへ書き込む

FlashCAN.exeの操作方法は、第8章を参照ください。

### 3.3 このアプリケーションノートでのCAN通信設定

このアプリケーションノートでは、CANの通信設定を下記のように設定します。

§ CANのボーレート：1000kbps

(CPU動作周波数20MHz, BCRレジスタの設定値=H'0034)

- BRP：設定値=0(1 time quantum / 2システムクロック)
- TSEG1：設定値=4(5 time quantum)
- TSEG2：設定値=3(4 time quantum)

よって、1Mbps = 20MHz / { 2 × ( 0 + 1 ) × ( 3 + 4 + 3 ) }

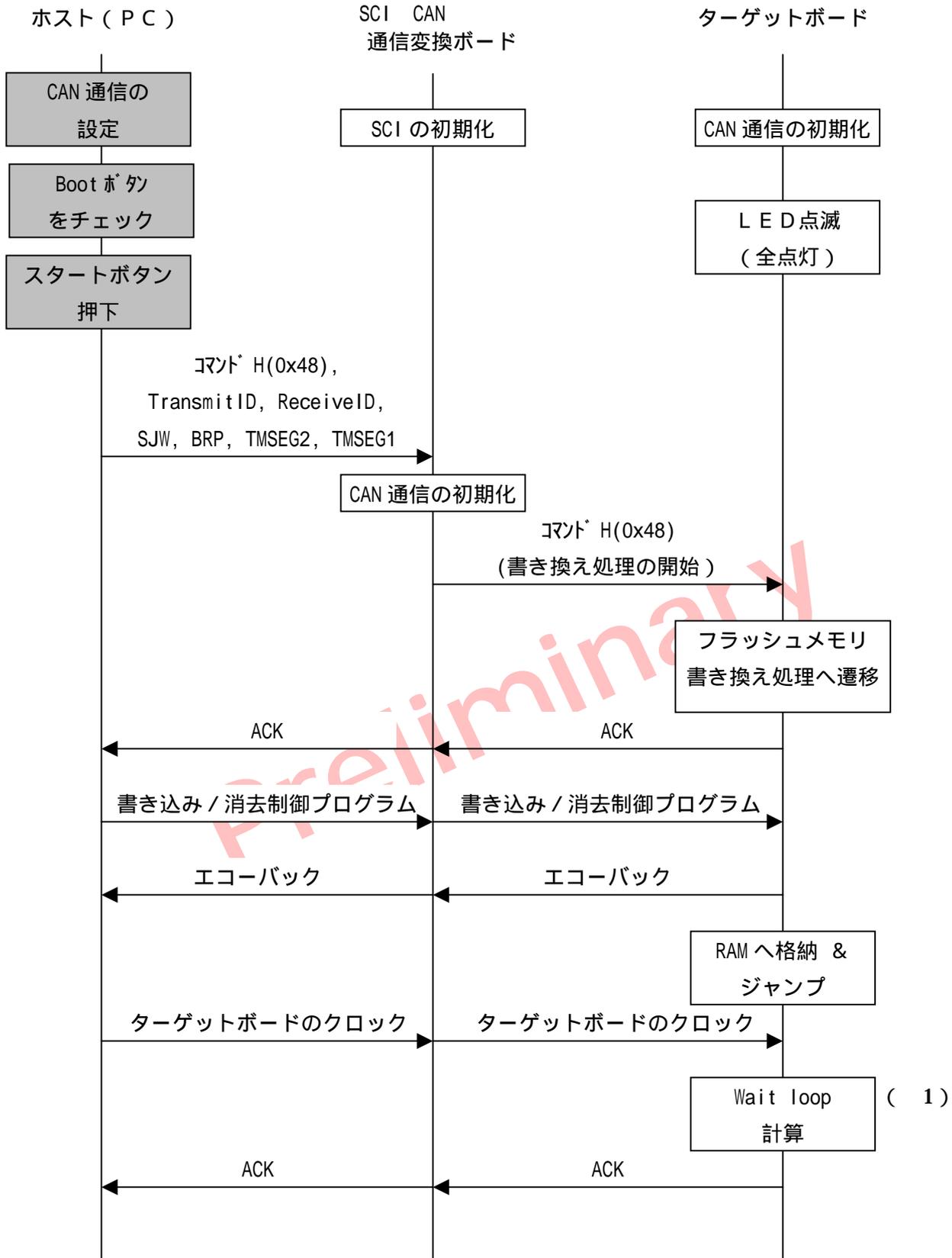
BRP                      TSEG1 TSEG2

§ データフレームのID (スタンダードフォーマット 11ビット)

- 通信変換ボード側の受信データ ID    : H'03F9 (11ビット表示：01111111001)
- ターゲットボード側の受信データ ID : H'0602 (11ビット表示：11000000010)

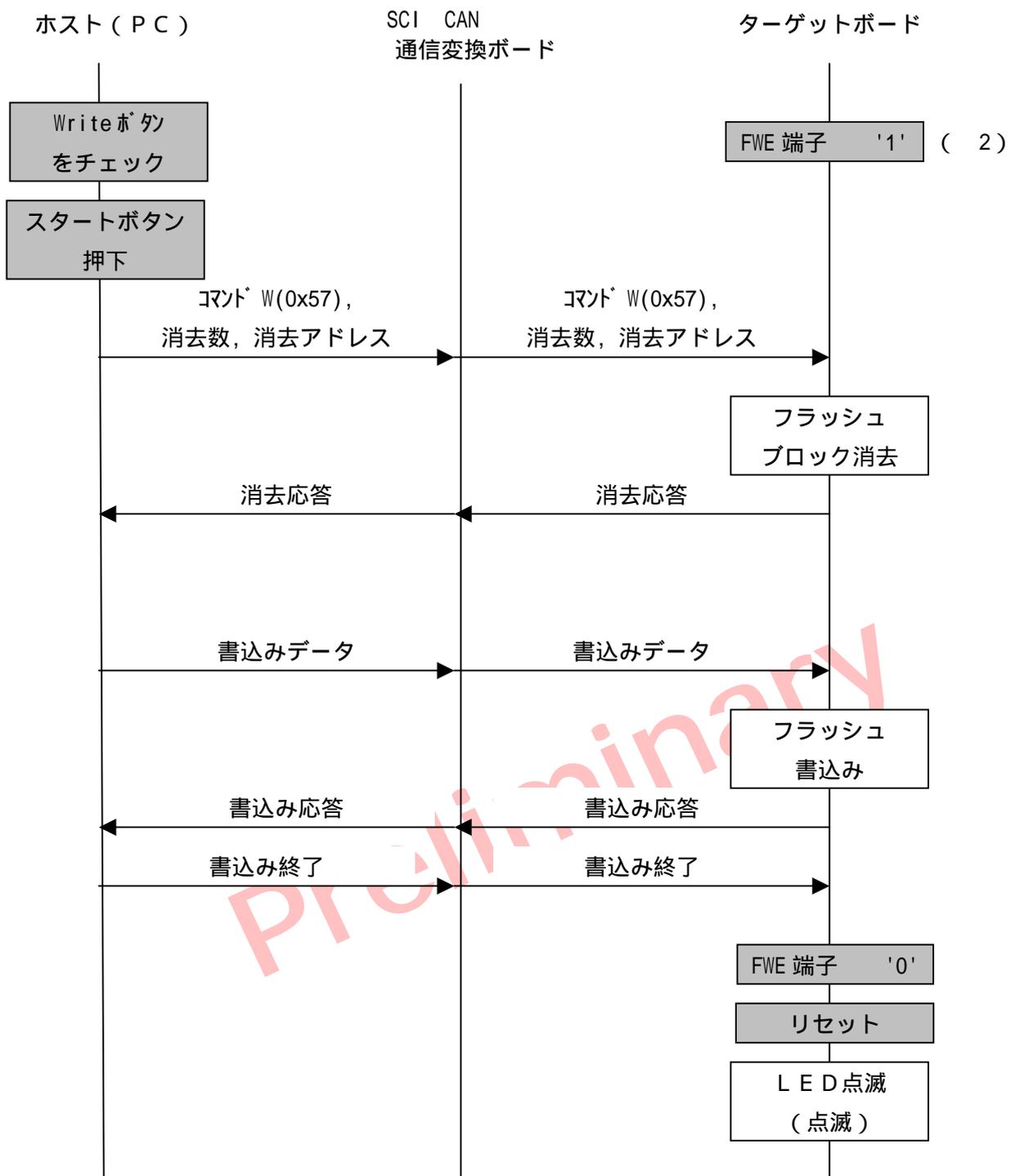
Preliminary

### 3.4 ユーザプログラムモードでのフラッシュメモリの書き換えシーケンス



網掛け部分はユーザの操作を示します。

( 1 ) 書き込み / 消去制御プログラムは、E ビット、P ビットの印加時間等をソフトウェアループの回数で制御しています。F-ZTAT マイコンの動作周波数に応じて、ループ回数を増減します。



( 2) 書き込みの「スタート」ボタンを押下する前に、FWE 端子を '1' にしてユーザプログラムモードへ遷移してください。

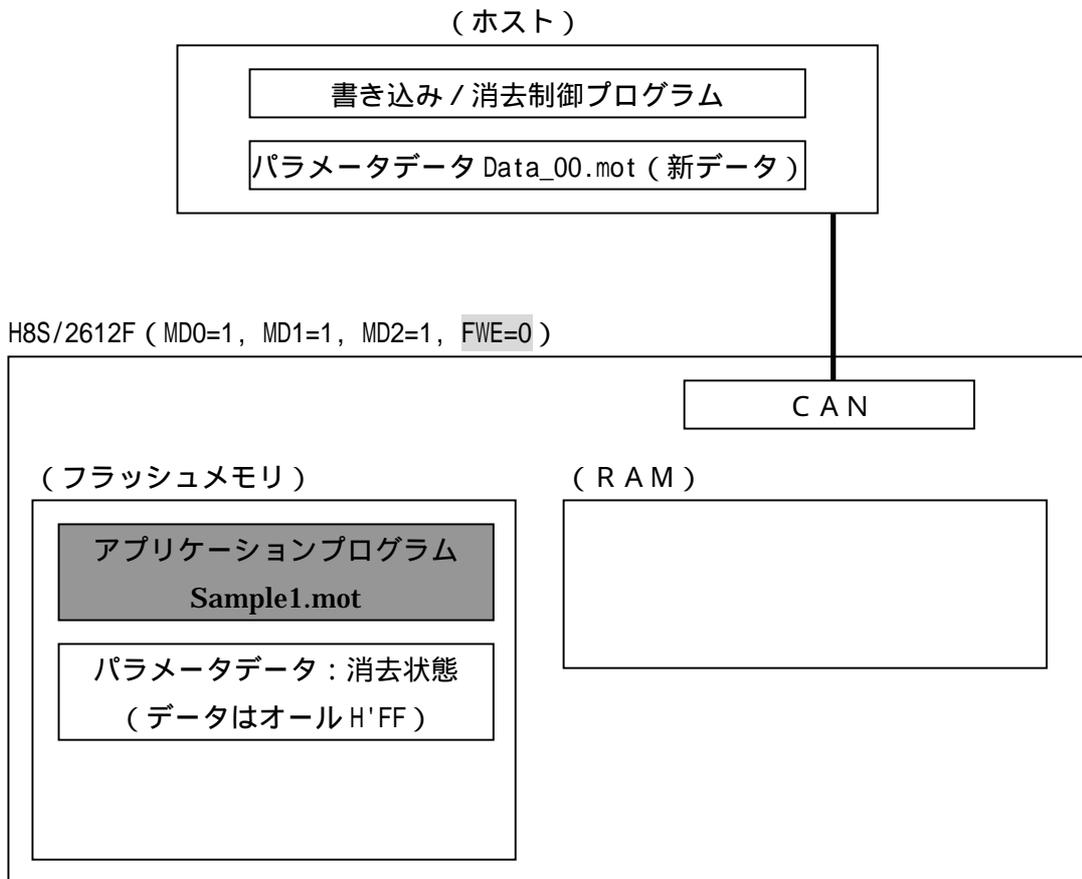
[ 備考 ]

Host ( PC ) と SCI CAN 通信変換ボード間の SCI 通信設定は、次の通りです。

- ・モード：調歩同期方式
- ・データフォーマット：8ビットデータ、パリティなし、1ストップビット
- ・ビットレート：57,600 bit / sec

## 4 . ユーザプログラムモードでのフラッシュメモリ書き換え動作の説明

### 4 . 1 初期状態

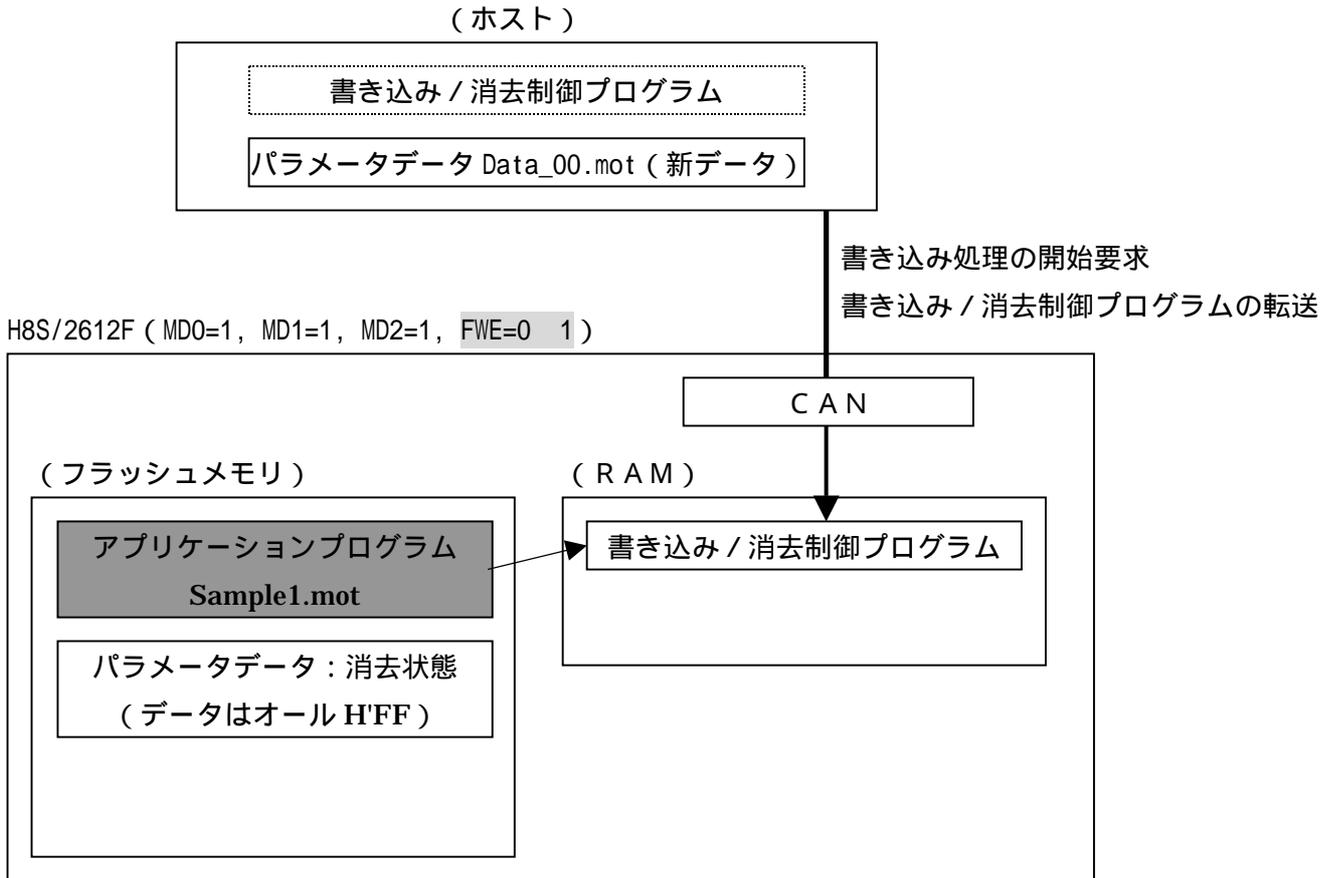


#### 説明

- (1) ユーザモード (MD0=1, MD1=1, MD2=1, FWE=0) で H8S/2612F をリセットスタートします。
- (2) アプリケーションプログラムは、書き込み処理への遷移条件を受け付けるために、HCAN ユニットを初期化します。
- (3) パラメータデータのエリアは、ブートモードにより消去されているので、データは H'FF となっています。このため、アプリケーションプログラムは、LED を全点灯します。

アプリケーションプログラムの LED 操作については、「5 . 3 アプリケーション (サンプル) プログラム」を参照してください。

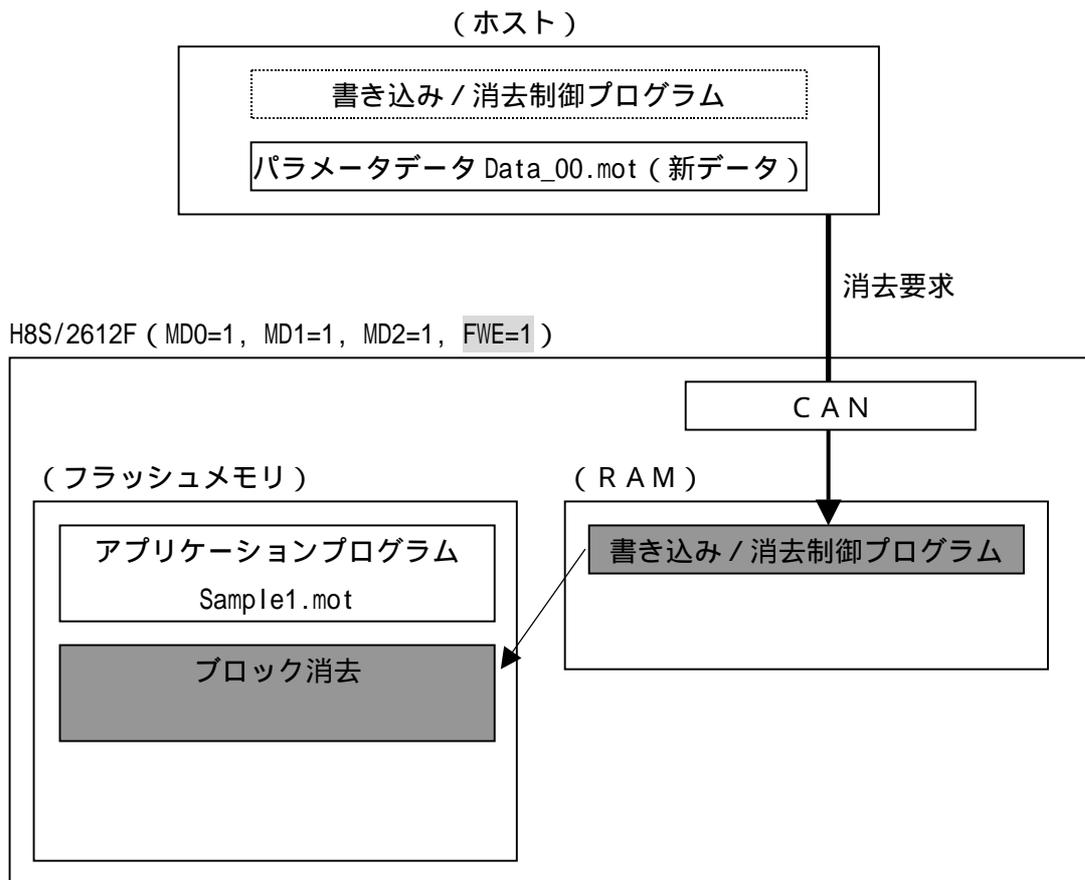
## 4.2 書き込み / 消去制御プログラムの転送



### 説明

- (1) ホストから HCAN 通信で書き換え処理の開始要求を発行します。
- (2) アプリケーションプログラムは、書き換え処理の開始要求を受け付けて、書き込み / 消去制御プログラムの転送処理へ遷移します。
- (3) ホストから書き込み / 消去制御プログラムを転送します。
- (4) アプリケーションプログラムは、転送される書き込み / 消去制御プログラムを RAM へ格納し、転送終了後、RAM 上の書き込み / 消去制御プログラムへジャンプします。
- (5) ユーザは、ターゲットボードのスイッチを操作し、FWE 端子を ON(1)に切り換えます。FWE=1 とすることでハードウェアプロテクトが解除され、フラッシュメモリの消去 / 書き込みが可能となります。

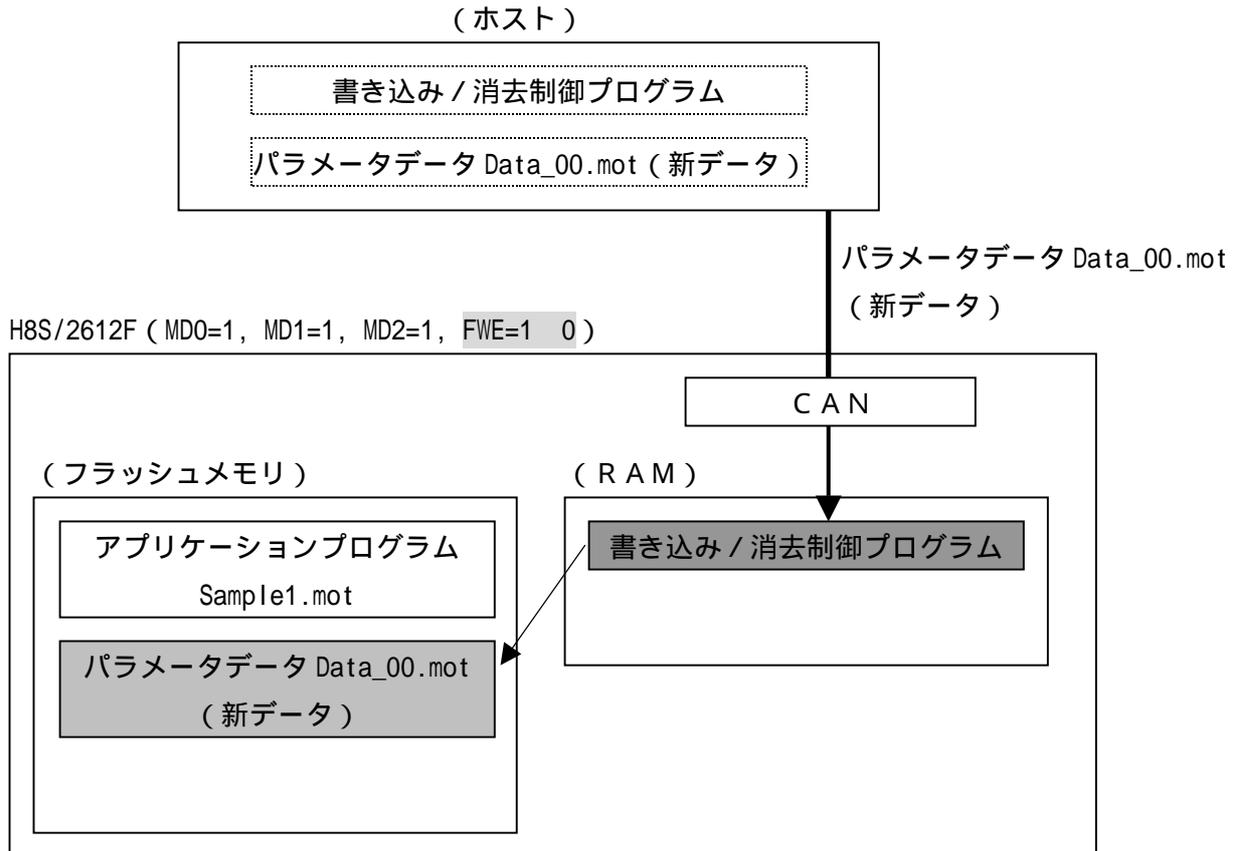
### 4.3 フラッシュメモリのブロック消去



#### 説明

- (1) ホストは HCAN 通信により、書き込み対象とするエリアのブロック消去要求を発行します。
  - (2) RAM 上の書き込み/消去制御プログラムは、要求のあったブロックを消去します。
- 但し、初期状態ではあらかじめ消去されているので、イレースベリファイのみ行ない、イレースは実施しません。

#### 4.4 新アプリケーションプログラムの書き込み



#### 説明

- (1) ホストは HCAN 通信で、書き込みデータ(Data\_00.mot)を供給します。
- (2) RAM 上の書き込み / 消去制御プログラムは、書き込みデータを受信してフラッシュメモリへ書き込みます。
- (3) 書き込み終了後、FWE 端子を OFF(0)してください。
- (4) リセットスタートすると、アプリケーションプログラムはパラメータデータ (新データ) を参照し、LED を点滅させます。

アプリケーションプログラムの LED 操作については、「5.3 アプリケーション (サンプル) プログラム」を参照してください。

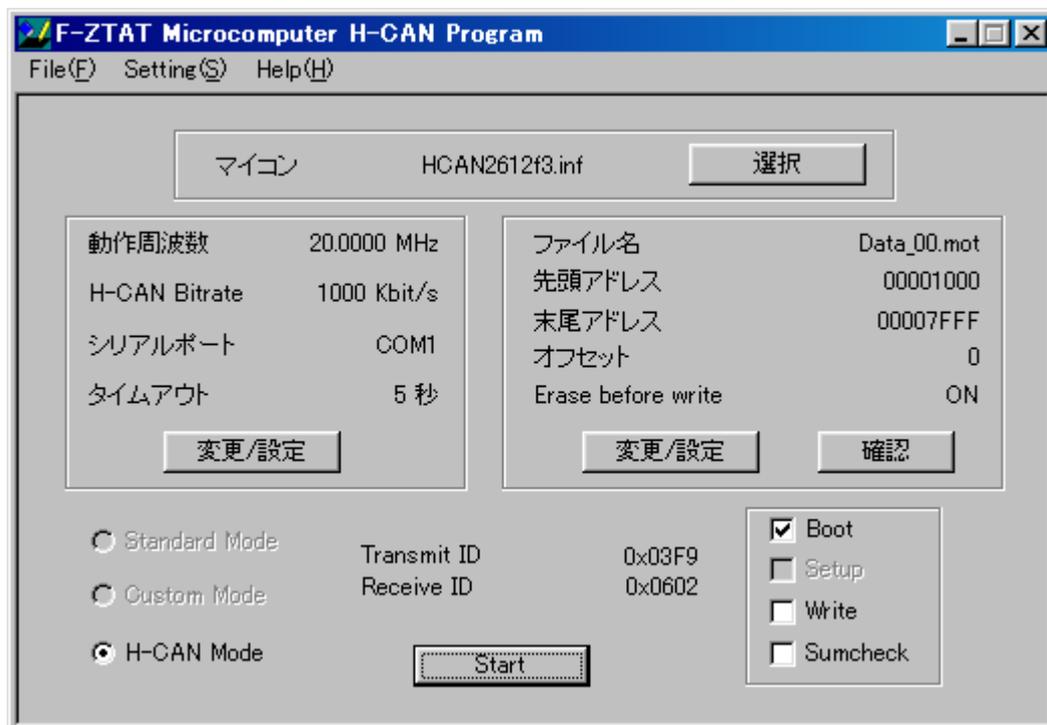
## 5 . ソフトウェアの詳細説明

### 5 . 1 F-ZTAT Microcomputer H-CAN Program

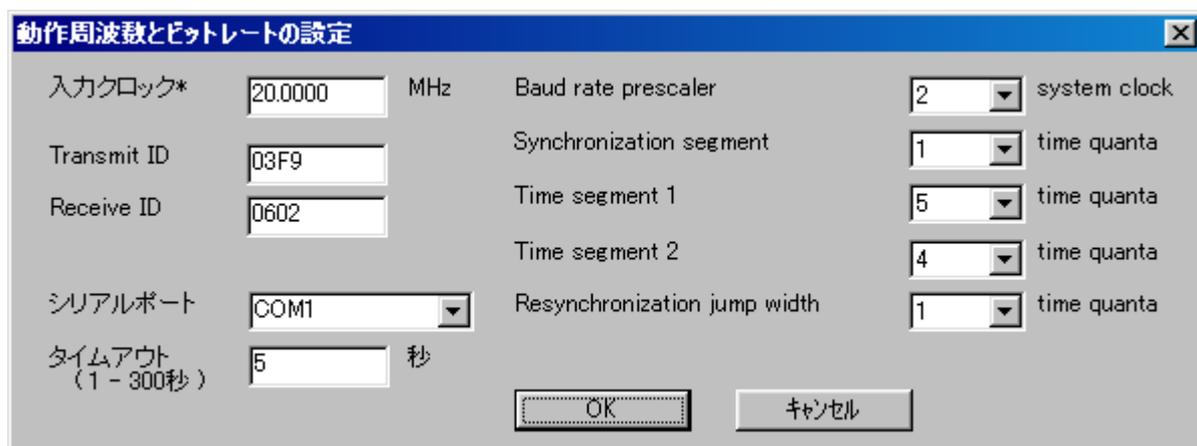
F-ZTAT Microcomputer H-CAN Program(FlashCAN.exe)は、ホスト(PC)で実行します。

	ファイル名称	内容
1	FlashCAN.exe	F-ZTAT Microcomputer H-CAN Programのプログラム本体です。
2	SCI2612F3.inf	H8S/2612Fマイコン固有の情報ファイル(SCI通信)です。
3	SCI2612F3.sub	「書き込み制御プログラム」のロードモジュールです。
4	HCAN2612F3.inf	H8S/2612Fマイコン固有の情報ファイル(H-CAN通信)です。
5	HCAN2612F3.sub	「書き込み/消去制御プログラム」のロードモジュールです。
6	HCAN2612F3.src	「書き込み/消去制御プログラム」のソースファイルです。
7	HCAN2612F3.bat	「書き込み/消去制御プログラム」をアセンブル、リンクするDOSプロンプト用バッチファイルです。
8	Hew2.hws	「書き込み/消去制御プログラム」をアセンブル、リンクするHewプロジェクトファイルです。

FlashCAN.exeの画面例 (メイン画面)



(CAN通信仕様の設定画面)



## 5.2 SCI HCAN 通信変換プログラム

SCI HCAN通信変換プログラム(SCItoCAN.mot)は、SCI HCAN通信変換ボードで実行します。

	ファイル名称	内容
1	SCItoCAN.mot	SCI HCAN通信変換ボード上のH8S/2612Fに内蔵されるロードモジュールです。 SCIとCANのインタフェースを変換します。 SCI HCAN通信変換ボード上のH8S/2612Fに内蔵されているSCIとHCANを使用します。

### 備考

SCI HCAN 通信変換プログラムは、ブートモードで SCI HCAN 通信変換ボードへダウンロードしておきます。

Preliminary

### 5.3 アプリケーション(サンプル)プログラム

アプリケーション(サンプル)プログラム(Sample1.mot)は、ターゲットボードで実行します。

	ファイル名称	内容
1	Sample1.mot	ターゲットボード上のH8S/2612Fに内蔵されるロードモジュールです。 ユーザプログラムモードでのフラッシュメモリの書き換え処理を組み込んでいます。 アプリケーションの動作を目視するために、サンプルとしてLEDを点灯/点滅させる機能としています。
2	Data_FF.mot Data_00.mot	Sample1.motが、LEDを点灯/点滅させるパターンのパラメータデータです。 ・H'FF(消去状態) : LEDを全点灯します。 ・H'00 : LEDを約1秒間隔で点滅します。
3	HCAN_up.c	Sample1.motのソースファイルです。 フラッシュメモリの書き換え処理を組み込んでいます。
4	Sample1.c	Sample1.motのソースファイルです。 LEDを点灯/点滅させる機能を組み込んでいます。
5	io_2612.h	内蔵I/Oレジスタを定義したインクルードファイルです。 HCAN_up.c / Sample.cからインクルードしています。
6	Sample1.bat	コンパイル、リンクのDOSプロンプト用バッチファイルです。
7	Hew2.hws	コンパイル、リンクのHewプロジェクトファイルです。

Preliminary

## 6 . アプリケーション ( サンプル ) プログラムの作成例

### 6 . 1 関数、変数一覧

#### ( 1 ) 関数一覧

ソース名 : HCAN\_up.c

略称	モジュール名	機能
PowerON_Reset	パワーオンリセット処理	初期化処理をする。 “H”コマンド(0x48)受信チェックをし、ユーザプログラム処理 / サンプルプログラムの切り替えをする。
UserProgramMode_Main	ユーザプログラム処理	書き込み / 消去制御プログラムの転送をする。
InitHCAN	HCANの初期化処理	HCAN通信の初期化をする。
CAN_MB5_trsr1byte	CAN1バイト送信処理	データを送信する。(HCAN通信)
CAN_MB4_rcvr1byte	CAN1バイト受信処理	データを受信する。(HCAN通信)

ソース名 : Sample1.c

UserApli	LED表示切り替え処理	パラメータチェックし、LEDの制御をする。 (全点灯/全点滅)
----------	-------------	------------------------------------

#### ( 2 ) 変数一覧

使用変数は、全て内部変数です。

### 6 . 2 CAN 通信設定の変更例

#### ( 1 ) ビットレートを変更。( 1000kbit/s 500kbit/s )

HCAN.BCR.WORD = 0x0034; を= 0x0134; に変更します。

Baud rate prescalor が 4system clock となりビットレートが 500kbit/s になります。

ハードウェアマニュアルのレジスタの説明(BCR)を参照してください。

#### ( 2 ) データ長を変更。( 1byte 8byte )

HCAN.MC[4][1-1] = 0x01; を= 0x08; に変更します。

受信データ長が 8 バイトになります。

HCAN.MC[5][1-1] = 0x01; を= 0x08; に変更します。

送信データ長が 8 バイトになります。

ハードウェアマニュアルのレジスタの説明(MC0 ~ 15)を参照してください。

#### ( 3 ) 受信メールアドレス ID を変更。( 0x03F9 0x0005 )

HCAN.MC[4][5-1] = 0x20; を= 0xA0; に変更します。

HCAN.MC[4][6-1] = 0x7F; を= 0x00; に変更します。

受信メールアドレス ID ( ID28 ~ 18 ) が 0x0005 になります。

#### ( 4 ) 送信メールアドレス ID を変更。( 0x0602 0x0004 )

HCAN.MC[5][5-1] = 0x40; を= 0x80; に変更します。

HCAN.MC[5][6-1] = 0xC0; を= 0x00; に変更します。

送信メールアドレス ID ( ID28 ~ 18 ) が 0x0004 になります。

ハードウェアマニュアルのレジスタの説明(MC0 ~ 15)を参照してください。

プログラムリスト ( 1 ) ~ ( 3 )

ソース名 : HCAN\_up.c

モジュール名 : InitHCAN

```

/*****/
/* H C A Nの初期化 */
/* ・ ボーレート : 1000kbps、 TSEG1=4, TSEG2=3, BRP=0, SJW=0 */
/* ・ MB4 : 受信「フラッシュメモリ書換え要求」 [ ID(11bit):03F9] */
/* ・ MB5 : 送信「フラッシュメモリ書換え応答」 [ ID(11bit):0602] */
/* [備考] RAM 上で動作中に通信するので、割込みは一切使用しない設定にします。 */
/*****/
void InitHCAN( void )
{
    int i, j;                /* ループカウンタ */
    /* H C A Nのモジュールストップビット(M S T P C R C )の解除 */
    System.MSTPCRC.BIT.HCANCKSTP = 0;
    HCAN.BCR.WORD = 0x0034; 0x0134; /* 1000kbps(at 20MHz)ボレーレート設定 */
    HCAN.MBCR.WORD = 0x1100; /* メールボックス送信 & 受信設定 MB4=Rcv */
    /* MC[4-5][0-7], MD[4-5][0-7]を全て0クリア */
    for ( j = 4; j <= 5; j++ ) {
        for ( i = 0; i < 8; i++ ) {
            HCAN.MC[j][i] = 0x00; /* MC[4][0] ~ MC[5][7] */
            HCAN.MD[j][i] = 0x00; /* MD[4][0] ~ MD[5][7] */
        }
    }
    HCAN.MCR.BYTE = 0x04; /* 送信方式 : MB 順 */
    while ( HCAN.GSR.BIT.GSR3 == 1 ); /* HCAN 通常モードでない間 */
    /* HCAN 通常モードへ遷移 */
    /* メールボックス 4 ( 受信 : フラッシュメモリ書換え要求 ) の設定 */
    HCAN.MC[4][1-1] = 0x01; 0x08; /* MB4 データ長 = 1 バイト */
    HCAN.MC[4][5-1] = 0x00; /* データフレーム、スタンダードフォーマット */
    HCAN.MC[4][5-1] = 0x20; 0xA0; /* ID: x xxxx xxx0 01-- ---- ---- ---- */
    HCAN.MC[4][6-1] = 0x7F; 0x00; /* ID: 0 1111 111x xx-- ---- ---- ---- */
    /* メールボックス 5 ( 送信 : フラッシュメモリ書換え応答 ) の設定 */
    HCAN.MC[5][1-1] = 0x01; 0x08; /* MB5 データ長 = 1 バイト */
    HCAN.MC[5][5-1] = 0x00; /* データフレーム、スタンダードフォーマット */
    HCAN.MC[5][5-1] = 0x40; 0x80; /* ID: x xxxx xxx0 10-- ---- ---- ---- */
    HCAN.MC[5][6-1] = 0xC0; 0x00; /* ID: 1 1000 000x xx-- ---- ---- ---- */
}

```

### 6.3 CAN 通信の受信 / 送信メールアドレスの変更例

受信 / 送信メールアドレスの変更。(受信 : [4] [14] / 送信 : [5] [15])

(1) メールボックスの送受信方向を変更。

HCAN.MBCR.WORD = 0x1100; を 0x0140; に変更します。

MBCR14 のみが 1 となりメールアドレス 14 のみ受信に設定されます。

ハードウェアマニュアルのレジスタの説明(MBCR)を参照してください。

(2) メールボックスの初期化処理を変更。

for ( j = 4; j <= 5; j++ ) を for ( j = 14; j <= 15; j++ ) { に変更します。

メールアドレス 14/15 のみ初期化処理をします。

ハードウェアマニュアルのレジスタの説明(MD0 ~ 15) / (MC0 ~ 15)を参照してください。

(3) メールボックスの設定を変更。

~ HCAN.MC[4] を HCAN.MC[14] に変更します。

~ HCAN.MC[5] を HCAN.MC[15] に変更します。

受信メールアドレス[14] / 送信メールアドレス[15]の設定をします。

ハードウェアマニュアルのレジスタの説明 (MC0 ~ 15)を参照してください。

(4) トリガ受信処理を変更。

if(HCAN.RXPR.BIT.RXPR4== 1){ を if(HCAN.RXPR.BIT.RXPR14== 1){ に変更します。

ハードウェアマニュアルのレジスタの説明 (RXPR)を参照してください。

(5) 受信処理を変更。

while( HCAN.RXPR.BIT.RXPR4 == 0 ); を while(HCAN.RXPR.BIT.RXPR14 == 0); に変更します。

HCAN.RXPR.WORD = 0x1000; を 0x0040; に変更します。

return HCAN.MD[4][0] を return HCAN.MD[14][0]; に変更します。

ハードウェアマニュアルのレジスタの説明 (RXPR) / (MD0 ~ 15)を参照してください。

(6) 送信処理を変更。

HCAN.MD[5][0] を HCAN.MD[15][0] に変更します。

HCAN.TXPR.BIT.TXPR5 を HCAN.TXPR.BIT.TXPR15 に変更します。

while( HCAN.TXPR.BIT.TXPR5 == 0 ); を while( HCAN.TXPR.BIT.TXPR15 == 0 ); に変更します。

HCAN.TXACK.WORD = 0x2000; を 0x0080; に変更します。

ハードウェアマニュアルのレジスタの説明(MD0 ~ 15) / (TXPR) / (TXACK)を参照してください。

プログラムリスト ( 1 ) ~ ( 3 )

ソース名 : HCAN\_up.c

モジュール名 : InitHCAN

```

/*****/
/* H C A Nの初期化 */
/* ・ボーレート : 1000kbps、TSEG1=4,TSEG2=3,BRP=0,SJW=0 */
/* ・MB4 : 受信「フラッシュメモリ書換え要求」[ID(11bit):03F9] */
/* ・MB5 : 送信「フラッシュメモリ書換え応答」[ID(11bit):0602] */
/* [備考] RAM上で動作中に通信するので、割込みは一切使用しない設定にします。*/
/*****/
void InitHCAN( void )
{
    int i, j;          /* ループカウンタ */
    /* H C A Nのモジュールストップビット(M S T P C R C )の解除 */
    System.MSTPCRC.BIT.HCANCKSTP = 0;
    HCAN.BCR.WORD = 0x0034;          /* 1000kbps(at 20MHz)ボーレート設定 */
    HCAN.MBCR.WORD = 0x1100; 0x0140; /* メールボックス送信&受信設定 MB4=Rcv */
    /* MC[4-5][0-7],MD[4-5][0-7]を全て0クリア */
    for ( j = 4; j <= 5; j++ ) { for ( j = 14; j <= 15; j++ ) {
        for ( i = 0; i < 8; i++ ) {
            HCAN.MC[j][i] = 0x00; /* MC[4][0] ~ MC[5][7] */
            HCAN.MD[j][i] = 0x00; /* MD[4][0] ~ MD[5][7] */
        }
    }
    HCAN.MCR.BYTE = 0x04;          /* 送信方式 : MB 順 */
    while ( HCAN.GSR.BIT.GSR3 == 1 ); /* HCAN 通常モードでない間 */
    /* HCAN 通常モードへ遷移 */
    /* メールボックス4 (受信 : フラッシュメモリ書換え要求) の設定 */
    HCAN.MC[4] [14] [1-1] = 0x01; /* MB4 データ長 = 1バイト */
    HCAN.MC[4] [14] [5-1] = 0x00; /* データフレーム、スタンダートフォーマット */
    HCAN.MC[4] [14] [5-1] = 0x20; /* ID: x xxxx xxx0 01-- ---- ---- ---- */
    HCAN.MC[4] [14] [6-1] = 0x7F; /* ID: 0 1111 111x xx-- ---- ---- ---- */
    /* メールボックス5 (送信 : フラッシュメモリ書換え応答) の設定 */
    HCAN.MC[5] [15] [1-1] = 0x01; /* MB5 データ長 = 1バイト */
    HCAN.MC[5] [15] [5-1] = 0x00; /* データフレーム、スタンダートフォーマット */
    HCAN.MC[5] [15] [5-1] = 0x40; /* ID: x xxxx xxx0 10-- ---- ---- ---- */
    HCAN.MC[5] [15] [6-1] = 0xC0; /* ID: 1 1000 000x xx-- ---- ---- ---- */
}

```

プログラムリスト ( 4 )

ソース名 : HCAN\_up.c  
 モジュール名 : PowerOn\_Reset

```

/*****/
/* パワーオンリセット処理 */
/* ユーザプログラムモード - トリガ待ち ( コマンド 'H' 受信 ) */
/* サンプルプログラム / ユーザプログラムモードの切り替え */
/*****/
#pragma section _BOOT /* セクション名称"P_BOOT" */
void PowerON_Reset(void){ /* パワーオンリセット(ハタ番号0)ハンドラ */
    volatile unsigned char LatchMDCR;
    unsigned char Data;
    LatchMDCR = System.MDCR.BYTE; /* MD2-MD0 のラッチ */
    /* 割込みの禁止 */
    set_imask_ccr(1);
    set_imask_exr(7);
    HCAN.IRR.WORD = 0x0100; /* IRRO を"1"ライトし"0"クリア */
    InitHCAN();
    /* ユーザプログラム */
    Data = 0x00;
    PORT.PDDDR.BYTE = 0xFF; /* LED初期化 */
    PORT.PDDR.BYTE = 0xFF;
    while(Data != 'H'){ /* コマンド 'H'? */
        if(HCAN.RXPR.BIT.RXPR4 == 1){ /* 受信データあり? */
            Data = CAN_MB4_rcv1byte();
        }else{
            UserApli(); /* サンプルプログラム */
        }
    }
    PORT.PDDR.BYTE = 0xFF; /* LED消灯 */
    UserProgramMode_Main();
    sleep(); /* (ここには来ません。) */
}
    
```

プログラムリスト ( 5 )

ソース名 : HCAN\_up.c  
モジュール名 : CAN\_MB4\_rcv1byte

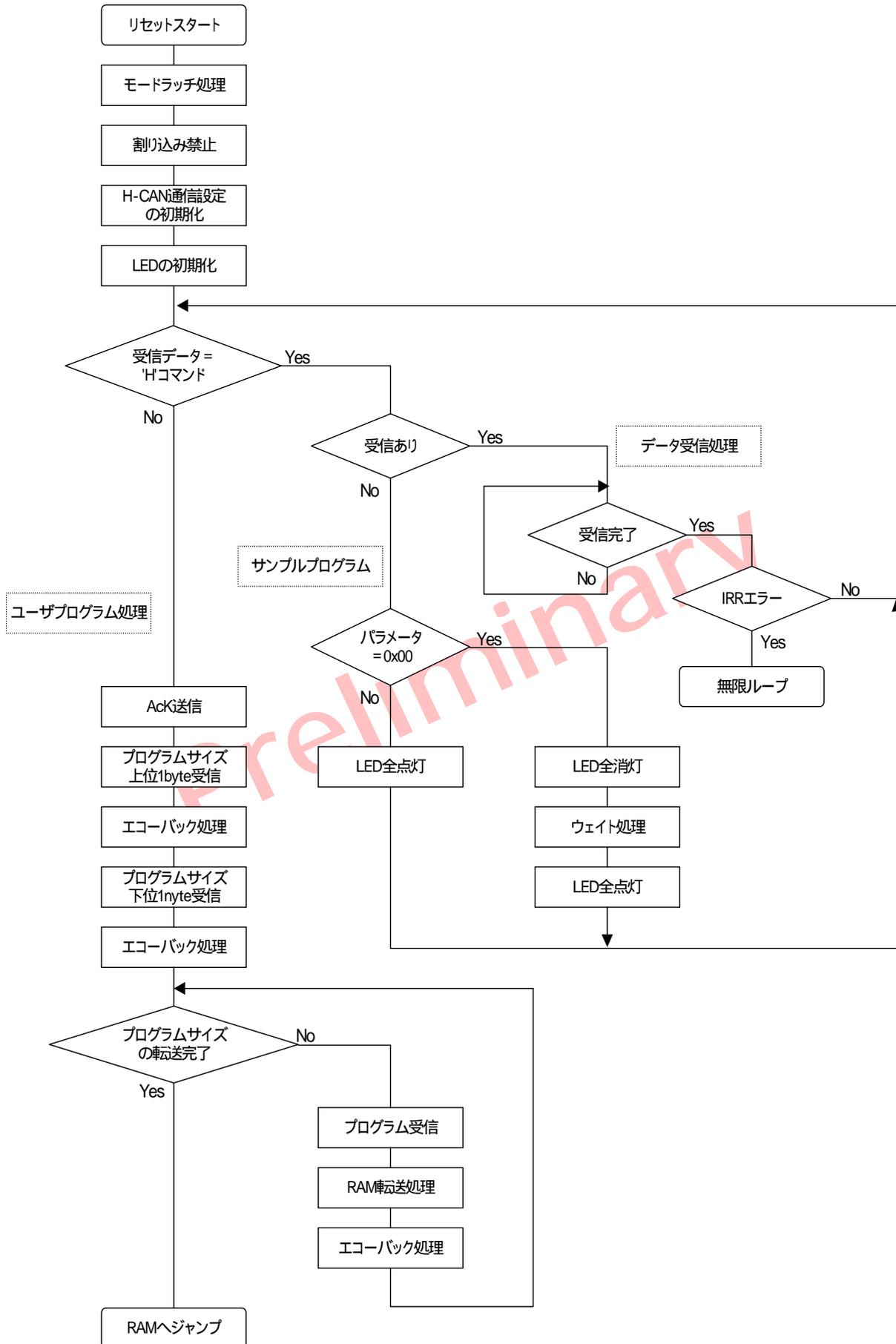
```
/* ***** */
/* CAN 1バイト受信 */
/* ***** */
unsigned char CAN_MB4_rcv1byte( void ){
    while( HCAN.RXPR.BIT.RXPR4 RXPR14 == 0 ); /* MB4 受信完了待ち */
        if( (HCAN.IRR.WORD & 0x1802) != 0 ){
            while(1); /* 無限ループ */
        }
    HCAN.RXPR.WORD = 0x1000; 0x0040; /* 受信フラグをクリア */
    return HCAN.MD[4] [14] [0]; /* MD4 から 1バイト受信 */
}
```

プログラムリスト ( 6 )

ソース名 : HCAN\_up.c  
モジュール名 : CAN\_MB5\_trs1byte

```
/* ***** */
/* CAN 1バイト送信 */
/* ***** */
void CAN_MB5_trs1byte( unsigned char TrsData ){
    HCAN.MD[5] [15] [0] = TrsData; /* MD5 に送信データ 1バイト設定 */
    HCAN.TXPR.BIT.TXPR5 TXPR15 |= 1; /* 送信開始 */
    while( HCAN.TXPR.BIT.TXPR5 TXPR15 == 0 ); /* 送信完了待ち */
        if( (HCAN.IRR.WORD & 0x1802) != 0 ){
            while(1); /* 無限ループ */
        }
    HCAN.TXACK.WORD = 0x2000; 0x0080; /* 送信完了 */
}
```

## 6.4 アプリケーション(サンプル)プログラムフローチャート





## (1) フラッシュメモリの消去

### ・イレース

イレースは、フラッシュメモリを消去する機能であり、1ブロック単位で行なう。イレースは、フラッシュメモリコントロールレジスタ(FLMCR)の SWE ビットを 1 にセット後、ブロック指定レジスタ(EBR)で消去するフラッシュメモリのエリアを 1 ビット設定し、FLMCR の ESU ビットをセットすることで、イレースモードへの準備(イレースセットアップ)を行い、FLMCR の E ビットをセットすることで、動作モードは、イレースモードへ遷移する。E ビットが設定されている時間が、消去時間である。消去時間経過後、FLMCR の E ビットのクリア、ESU ビットのクリア、SWE ビットのクリアでイレースモードを解除する。

### ・イレースベリファイ

イレースベリファイは、フラッシュメモリの消去が正常に消去されているかを検証する機能である。イレースベリファイは、FLMCR の SWE ビットのセット、次に EV ビットをセットすることで、動作モードはイレースベリファイモードへ遷移する。イレースベリファイモードでは、読み出す前に読み出すアドレスにデータ(H'FF)をダミーライトする。その後フラッシュメモリをリード(ベリファイデータは 16 ビットで読み出す)するとラッチしたアドレスのデータが読み出される。読み出しデータが消去(データがすべて 1)されていた場合、次のデータをダミーライトしイレースベリファイを行なう。ベリファイ完了後、FLMCR の EV ビットのクリア、SWE ビットのクリアでイレースベリファイモードを解除する。

下記に、フラッシュメモリ消去での FLMCR レジスタの各ビットのウェイト時間を示す。

**表 3 フラッシュメモリ消去で FLMCR レジスタの各ビットのウェイト時間**

各ビットのセット / クリア	ウェイト時間 (基準値)	ウェイト時間 20MHz
SWE セット	1 $\mu$ sec 以上	1.2 $\mu$ sec
ESU セット	100 $\mu$ sec 以上	100.2 $\mu$ sec
E セット	10 msec 以内	10.0 msec
E クリア	10 $\mu$ sec 以上	10.2 $\mu$ sec
ESU クリア	10 $\mu$ sec 以上	10.2 $\mu$ sec
EV セット	20 $\mu$ sec 以上	20.2 $\mu$ sec
ダミーライト	2 $\mu$ sec 以上	2.2 $\mu$ sec
EV クリア	4 $\mu$ sec 以上	4.2 $\mu$ sec
SWE クリア	100 $\mu$ sec 以上	100.2 $\mu$ sec
最大回数	100 回	100 回

## (2) フラッシュメモリの書き込み

### ・プログラム

プログラムは、フラッシュメモリに書き込む機能であり、1回の書き込みは128バイト単位で行う。プログラムは、フラッシュメモリコントロールレジスタ(FLMCR)のSWEビットを1にセットした後、128バイトの書き込みデータを書き込みデータエリアと再書き込みデータエリアに格納し、書き込むアドレス(書き込む先頭アドレスの下位8ビットは、H'00,H'80でなければならない)にRAM上の書き込みデータエリアの128バイトのデータを連続で書き込む(データ転送は、バイト単位で128連続して行なう)。フラッシュメモリはプログラムアドレスとプログラムデータをそれぞれフラッシュメモリにラッチする。

128バイト以下の書き込みでも128バイトのデータを転送を行なう必要があり、必要のないアドレスへの書き込みは、データをH'FFにして書き込む必要がある。次にFLMCRのPSUビットをセットすることで、プログラムモードへの準備(プログラムセットアップ)を行い、FLMCRのPビットをセットすることで、動作モードは、プログラムモードへ遷移する。

Pビットセットされている時間がフラッシュメモリの書き込み時間である。書き込み時間経過後、FLMCRのPビットのクリア、PSUビットのクリア、SWEビットのクリアでプログラムモードを解除する。(初期書き込み6回までは書き込み完了したビットに対して追加書き込みを行う。)

### ・プログラムベリファイ

プログラムベリファイは、フラッシュメモリの書き込みが正常に書かれているかを検証するモードであり、プログラムベリファイは、FLMCRのSWEビットをセット、次にPVビットをセットすることで、動作モードはプログラムベリファイモードへ遷移する。

プログラムベリファイモードでは、読み出す前に読み出すアドレスにデータH'FFをダミーライトする。その後フラッシュメモリを読み出す(ベリファイデータは16ビットで読み出す)とラッチしたアドレスのデータが読み出される。次に書き込みデータとベリファイデータを比較し再書き込みデータを演算し、再書き込みデータを再書き込みデータエリアに転送する。128バイト分のデータのベリファイが完了後、FLMCRのPVビットのクリア、SWEビットのクリアでプログラムベリファイモードを解除する。

下記に、フラッシュメモリ書き込みでFLMCRの各ビットのウェイト時間を示す。(初期書き込み6回までは追加書き込みデータを演算し追加書き込みデータエリアに転送する。)

**表4 フラッシュメモリ書き込みでFLMCRレジスタの各ビットのウェイト時間**

各ビットのセット /クリア	ウェイト時間 基準値	ウェイト時間 20MHz
SWE セット	1 $\mu$ sec 以上	1.2 $\mu$ sec
PSU セット	50 $\mu$ sec 以上	50.2 $\mu$ sec
P セット(1~6回)	30 $\mu$ sec 以内	30.0 $\mu$ sec
(1~6回)追加	10 $\mu$ sec 以内	10.0 $\mu$ sec
(7~1000回)	200 $\mu$ sec 以内	200.0 $\mu$ sec
P クリア	5 $\mu$ sec 以上	5.2 $\mu$ sec
PSU クリア	5 $\mu$ sec 以上	5.2 $\mu$ sec
PV セット	20 $\mu$ sec 以上	20.2 $\mu$ sec
ダミーライト	2 $\mu$ sec 以上	2.2 $\mu$ sec
PV クリア	2 $\mu$ sec 以上	2.2 $\mu$ sec
SWE クリア	100 $\mu$ sec 以上	100.2 $\mu$ sec
最大回数	1000回	1000回

## 7.2 関数、変数及び、定数一覧

### (1) 関数一覧

ソース名：HCAN2612f3.src

略称	モジュール名	機能
MAIN	メイン処理	スタックの初期化 ターゲットクロックの受信 / チェック コマンド受信 / チェック (W / Cコマンド)
WLOOP_INI	ウェイトループ 初期化処理	ウェイトループの初期化
WAITLOOP_CAL	ウェイトループ 計算メイン処理	ウェイトループの計算処理 / 設定
WLOOP_CAL	ウェイトループ 計算処理	ウェイトループの計算
WCMD	Wコマンド処理	消去処理 書き込み処理 チェックサム処理
GET_EADR	消去アドレス受信 処理	消去ブロック先頭アドレスの受信
BLK_CHECK	指定ブロックチェック 処理	指定ブロックのチェック
GET_WADR	書き込みアドレス受信 処理	書き込みアドレスの受信 (4バイト)
GET_BUFFER	書き込みデータ受信 処理	書き込みデータの受信 (128バイト)
RCVNBYTE	Nバイト受信処理	データNバイトの受信
RCV1BYTE	1バイト受信処理	データ1バイトの受信
TRS1BYTE	1バイト送信処理	データ1バイトの送信
XON_CHECK	XONチェック処理	書き込み処理時の応答チェック
FWRITE128	フラッシュ128バイト 書き込み処理	初期書き込みベリファイ 初期書き込み / 書き込みベリファイ 追加書き込み 再書き込み / 書き込みベリファイ (書き込み前) (1~6回) (1~6回) (7~1000回)
FWRITEVF	書き込みベリファイ 処理	書き込みベリファイ 再書き込みデータ作成 追加書き込みデータ作成
FWRITE	フラッシュ書き込み 処理	フラッシュの書き込み
BLK1_ERASE	1ブロック消去 処理	指定ブロックのチェック 初期消去ベリファイ 消去 / 消去ベリファイ (消去前) (1~100回)
FERASEVF	消去ベリファイ処理	消去ベリファイ
FERASE	フラッシュ消去処理	フラッシュの消去
CHECKSUM	チェックサム処理	チェックサム値の計算 / 送信 (4バイト)

## (2) 変数一覧

略称	変数名	説明	サイズ
W_ADR	書き込みアドレス	書き込みアドレスを格納	4バイト
W_BUF	書き込みバッファ	書き込みデータを格納	128バイト
BUFF	バッファ	再書き込みデータを格納	128バイト
OWBUFF	追加書き込みバッファ	追加書き込みデータを格納	128バイト
COUNT	カウンタ	消去回数のカウンタ 書き込み回数のカウンタ	2バイト
EVF_ST	消去先頭アドレス	消去ブロックの先頭アドレスを格納	4バイト
EVF_ED	消去末尾アドレス	消去ブロックの末尾アドレスを格納	4バイト
BLK_NO	消去指定ブロック	消去指定ブロック を格納	1バイト
VF_RET	ベリファイ結果フラグ	消去ベリファイ結果 書き込みベリファイ結果	1バイト
RESTSIZE	書き込みデータサイズ	書き込みデータサイズを格納	4バイト
E_ADR	消去ブロックアドレス	消去ブロックアドレスを格納	64バイト
E_ADR_PTR	消去ブロックアドレス ポインタ	消去ブロックアドレスのポインタ	4バイト
WORKCLK	クロック	ターゲットクロックを格納	4バイト
ERASEBLOCK	消去ブロック数	消去ブロック数を格納	1バイト
WLOOP1	ウェイト 1 $\mu$	1 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP2	ウェイト 2 $\mu$	2 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP4	ウェイト 4 $\mu$	4 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP5	ウェイト 5 $\mu$	5 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP10	ウェイト 10 $\mu$	10 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP20	ウェイト 20 $\mu$	20 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP50	ウェイト 50 $\mu$	50 $\mu$ secウェイトのループ回数を格納	2バイト
WLOOP100	ウェイト100 $\mu$	100 $\mu$ secウェイトのループ回数を格納	2バイト
WTIME10	書き込みウェイト 10 $\mu$	追加書き込み時、10 $\mu$ secウェイト のループ回数を格納	4バイト
WTIME30	書き込みウェイト 30 $\mu$	初期書き込み時、30 $\mu$ secウェイト のループ回数を格納	4バイト
WTIME200	書き込みウェイト 200 $\mu$	再書き込み時、200 $\mu$ secウェイト のループ回数を格納	4バイト
WTIME1000	消去ウェイトループ 10m	消去時、10msecウェイト のループ回数を格納	4バイト

### 7.3 CAN 通信の受信/送信メールアドレス番号の変更例

受信/送信メールアドレス番号の変更。(受信:[4] [14]/送信:[5] [15])

#### (1) レジスタ/ビットを変更。

~ の各レジスタ/ビットの定義を追加する。

```
RXPR14_MOV .EQU H'0040
RXPR_L .EQU H'FFF80F
RXPR14 .EQU 6
MD14_0 .EQU H'FFF920
MD15_0 .EQU H'FFF928
TXPR_L .EQU H'FFF807
TXPR15 .EQU 7
TXACK15 .EQU 7
TXACK15_MOV .EQU H'0080
```

ハードウェアマニュアルのレジスタの説明 (RXPR)/(TXPR)/(TXACK)/(MD0~15)

を参照してください。

#### (2) 受信処理を変更。

**RXPR4** を **RXPR14** に変更します。

RXPR を RXPR\_L に変更します。

MD4\_0 を MD14\_0 に変更します。

RXPR4\_MOV を RXPR14\_MOV に変更します。

ハードウェアマニュアルのレジスタの説明 (RXPR)/(MD0~15)を参照してください。

#### (3) 送信処理を変更。

**MD5\_0** を **MD15\_0** に変更します。

TXPR を TXPR\_L に変更します。

TXPR5 を TXPR15 に変更します。

TXACK5\_MOV を TXACK15\_MOV に変更します。

ハードウェアマニュアルのレジスタの説明 (TXPR)/(TXACK)/(MD0~15)

を参照してください。

プログラムリスト ( 1 )

ソース名 : HCAN2612f3.src

モジュール名 : データ

; HCAN			
RXPR	.EQU	H'FFF80E	; 受信完了レジスタ ( 16ビット)
RXPR4:	.EQU	4	
RXPR4_MOV:	.EQU	H'1000	
RXPR14_MOV	.EQU	H'0040	
RXPR_L	.EQU	H'FFF80F	; 受信完了レジスタ(下位 8ビット)
RXPR14	.EQU	6	
MD4_0	.EQU	H'FFF8D0	
MD5_0	.EQU	H'FFF8D8	
MD14_0	.EQU	H'FFF920	
MD15_0	.EQU	H'FFF928	
TXPR	.EQU	H'FFF806	
TXPR5:	.EQU	5	
TXPR_L	.EQU	H'FFF807	; 送信待ちレジスタ(下位 8ビット)
TXPR15	.EQU	7	
TXACK	.EQU	H'FFF80A	; 送信アクノレッジレジスタ
TXACK5:	.EQU	5	
TXACK5_MOV:	.EQU	H'2000	
TXACK15	.EQU	7	
TXACK15_MOV	.EQU	H'0080	
IRR	.EQU	H'FFF812	
IRR_ERR:	.EQU	H'1802	

プログラムリスト ( 2 )

ソース名 : HCAN2612f3.src

モジュール名 : RCV1BYTE

```

; *****
; * TITLE      / H-CAN 1 BYTE DATA RECEPTION      *
; * FUNCTION   / RECEIVE 1 BYTE DATA              *
; * INPUT      / -                                  *
; * OUTPUT     / R2L = RECEIVED DATA              *
; *****
RCV1BYTE.EQU  $
              SUB.W  RO,RO
              BLD.B  #RXPR4,@RXPR  #RXPR14,@RXPR_L
              BST.B  #RXPR4  #RXPR14,ROL
              MOV.W  RO,RO
              BEQ    RCV1BYTE
;
              MOV.W  @IRR,RO                ; ERROR CHECK
              AND.W  #IRR_ERR,RO
              BNE    RCV_ERR
;
              MOV.B  @MD4_0  @MD14_0,R2L    ; RECEIVE DATA TO ROH
;
              MOV.W  #RXPR4_MOV  #RXPR14_MOV,RO
              MOV.W  RO,@RXPR                ; RXPR4 CLEAR
              RTS
;
RCV_ERR BRA    RCV_ERR                ; INFINITE LOOP
;

```

プログラムリスト ( 3 )

ソース名 : HCAN2612f3.src

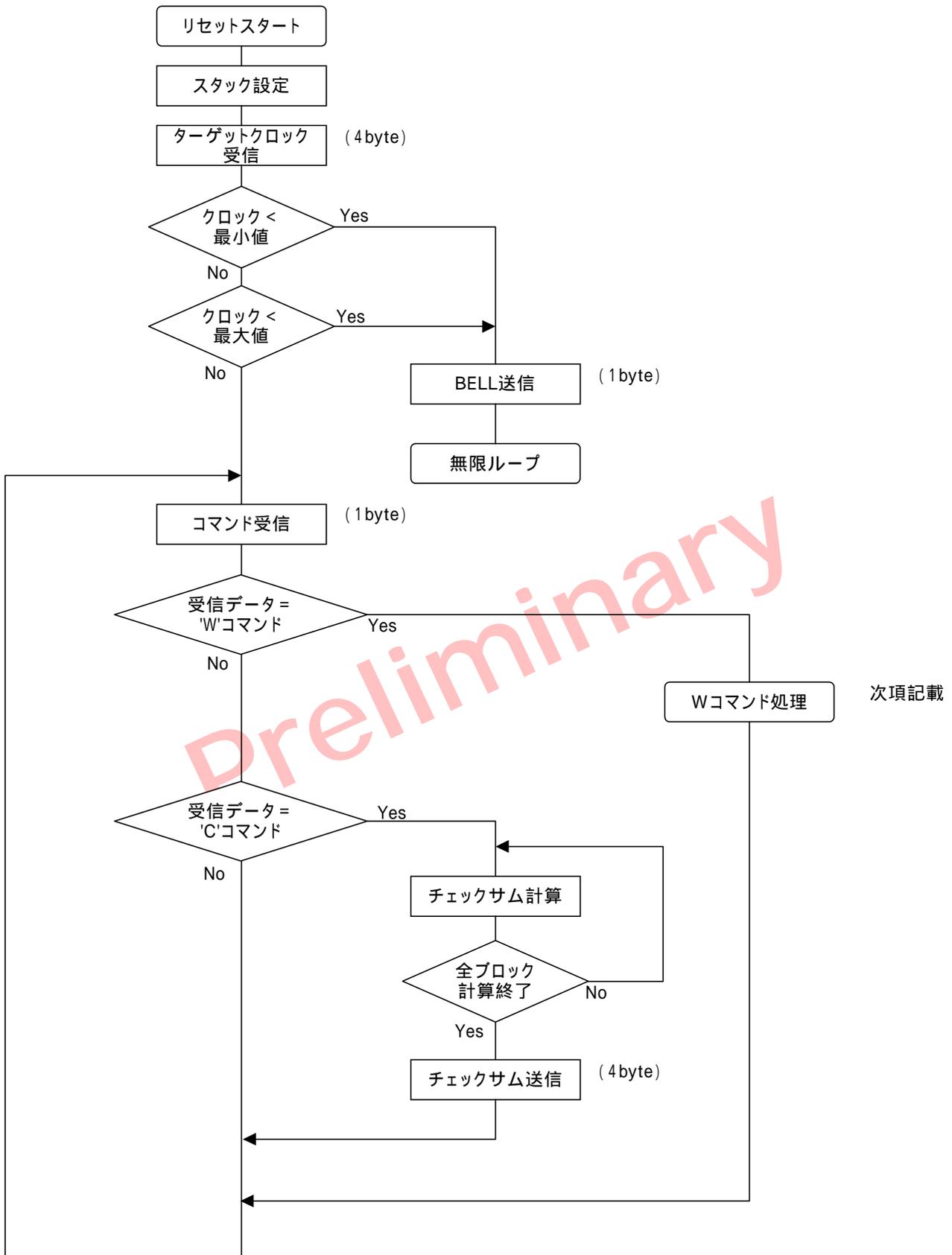
モジュール名 : TRS1BYTE

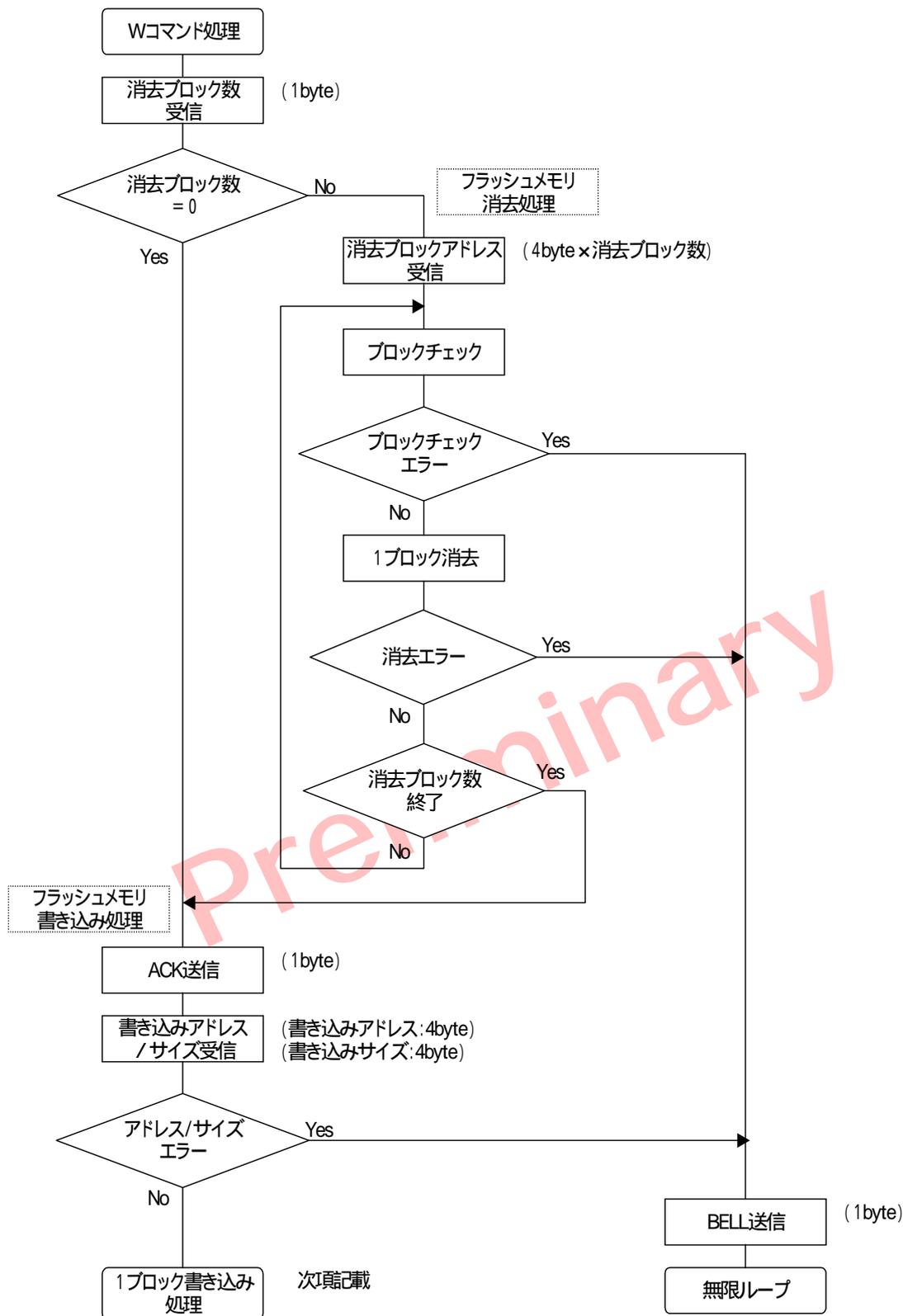
```

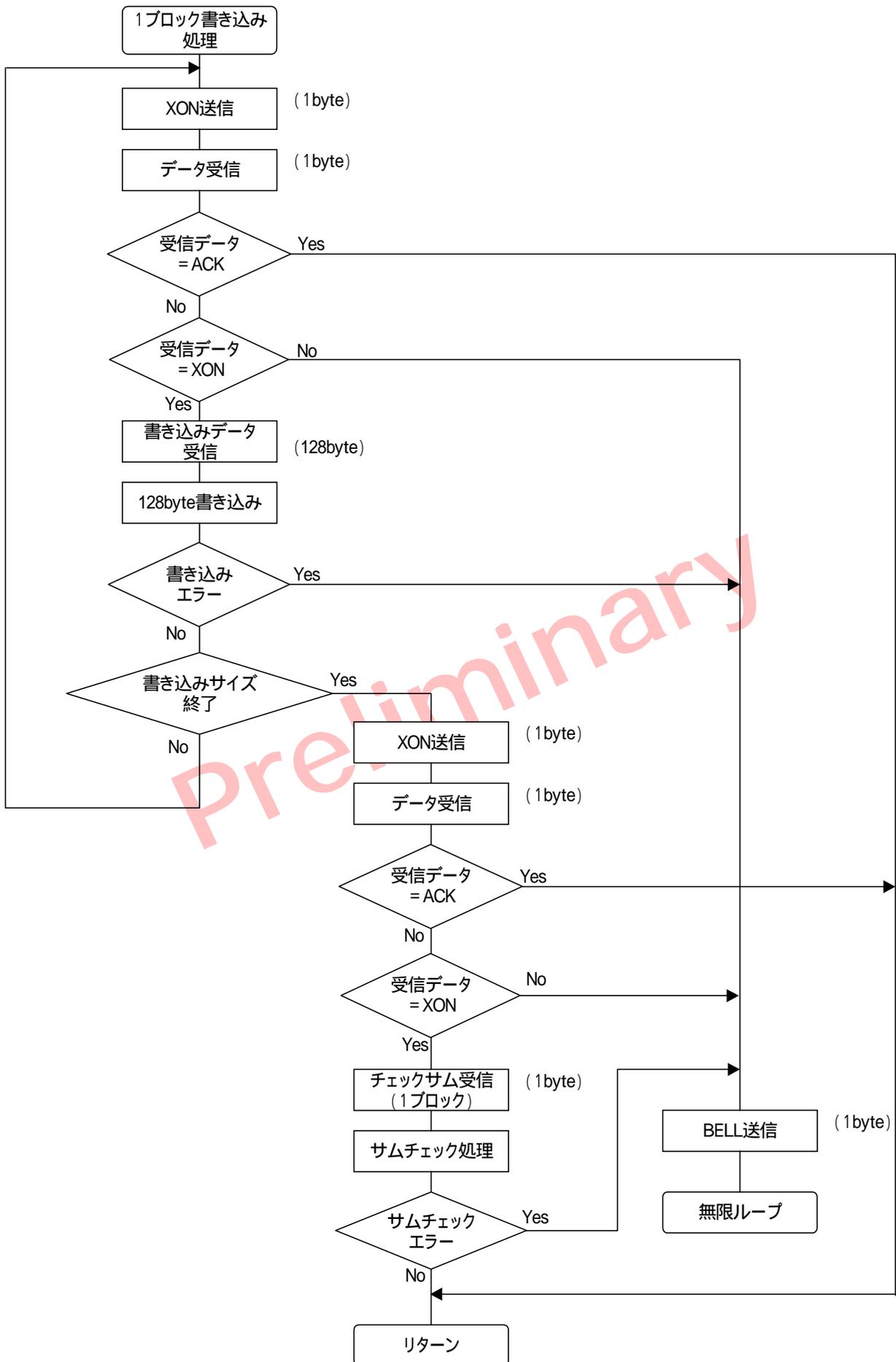
; *****
; * TITLE      / H-CAN 1 BYTE DATA TRANSMISSION      *
; * FUNCTION   / SEND 1 BYTE DATA                    *
; * INPUT      / R2L = SEND DATA                      *
; * OUTPUT     / -                                     *
; *****
TRS1BYTE.EQU    $
MOV.B    R2L,@MD5_0 @MD15_0 ; TRANSMIT R2L DATA TO MD5_0
;
MOV.W    @TXPR @TXPR_L,RO
BSET.B   #TXPR5 #TXPR15,ROH
MOV.W    RO,@TXPR @TXPR_L ; SET TXPR5
;
TRS_WAITSUB.W  RO,RO
BLD.B    #TXPR5, #TXPR15 @TXPR
BST.B    #TXPR5, #TXPR15 ROH
MOV.W    RO,RO
BNE     TRS_WAIT
;
MOV.W    @IRR,RO ; ERROR CHECK
AND.W    #IRR_ERR,RO
BNE     TRS_ERR
;
MOV.W    #TXACK5_MOV #TXACK15_MOV,RO
MOV.W    RO,@TXACK ; CLEAR TXACK5
RTS
;
TRS_ERR  BRA     TRS_ERR ; INFINITE LOOP
;

```

## 7.4 書き込み/消去プログラムフローチャート



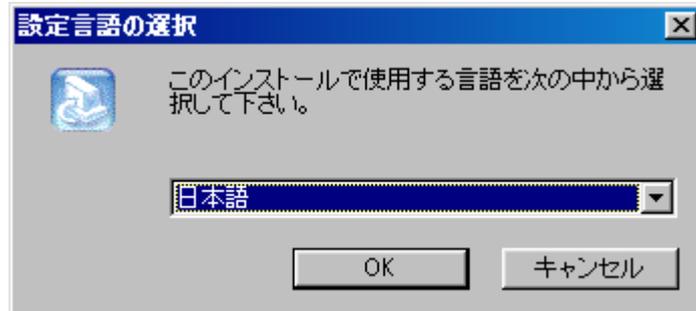




8 . オンボード書き込みツール及び、SCI HCAN 通信変換プログラムの機能及び、操作説明  
オンボード書き込みツールの詳細については、「F - Z T A Tマイコン オンボード書き込みプログラム マニユ  
アル」の各章を合わせてご参照ください。

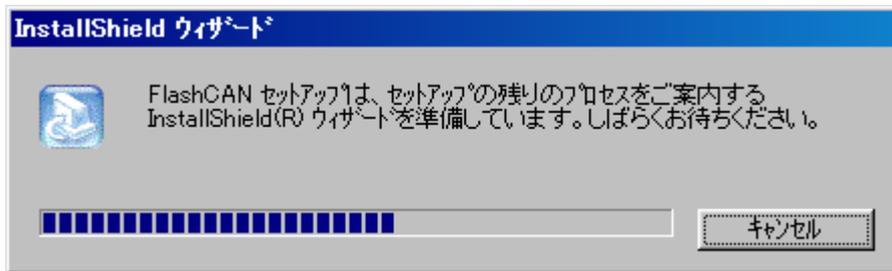
8 . 1 オンボード書き込みツールのインストール

手順(1) : Setup.exe を起動します。



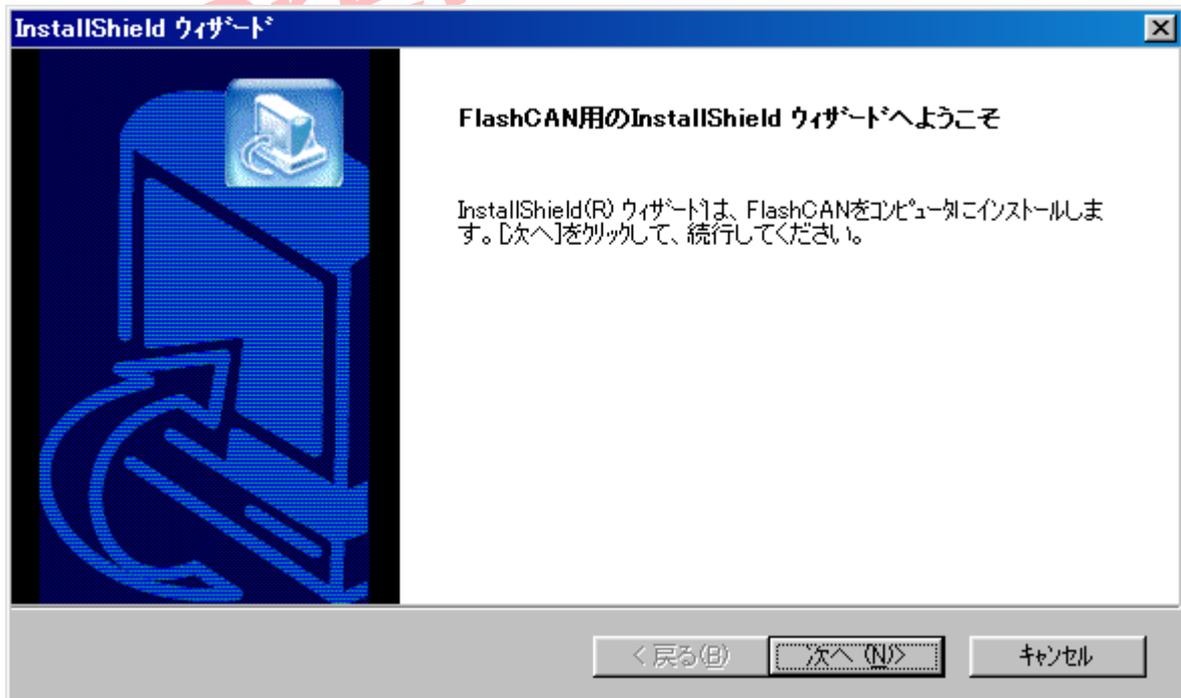
手順(2) : 設定言語を選択し、OK ボタンを押下します。

(キャンセルボタンを押下するとインストールを中止します。)



(キャンセルボタンを押下するとインストールを中止します。)

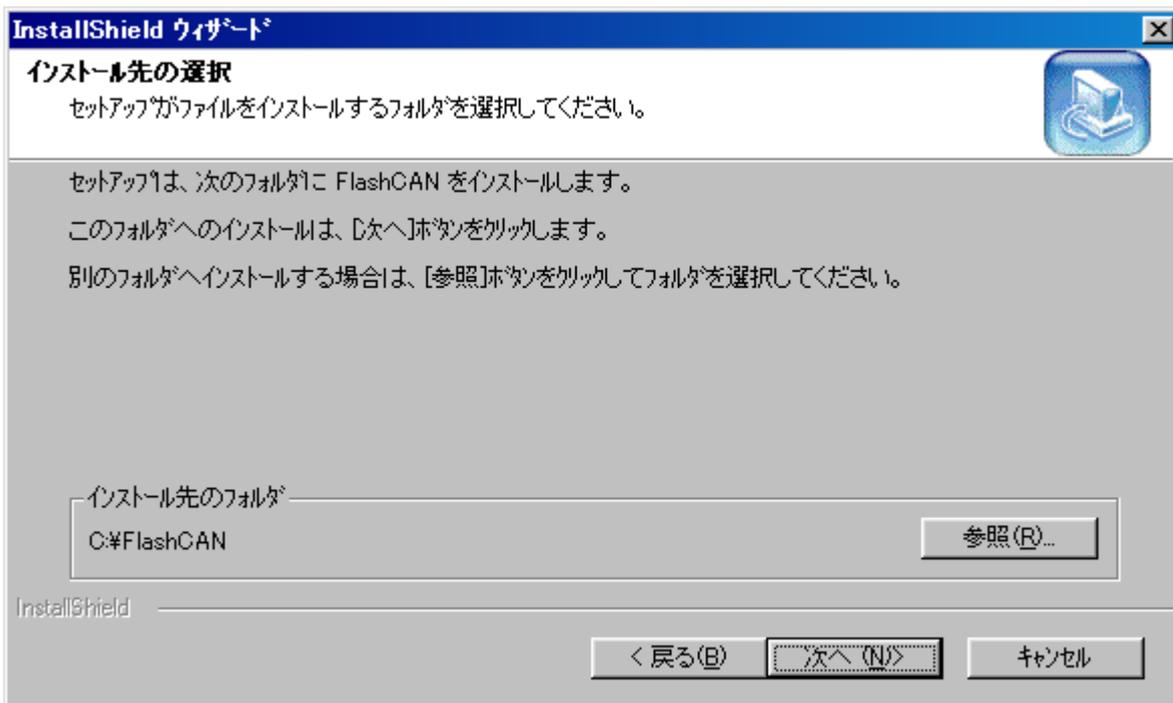
手順(3) : 次へボタンを押下し、インストールを続行します。



(キャンセルボタンを押下するとインストールの中止確認ダイアログを表示します。)

手順(4) : インストール先の選択をします。

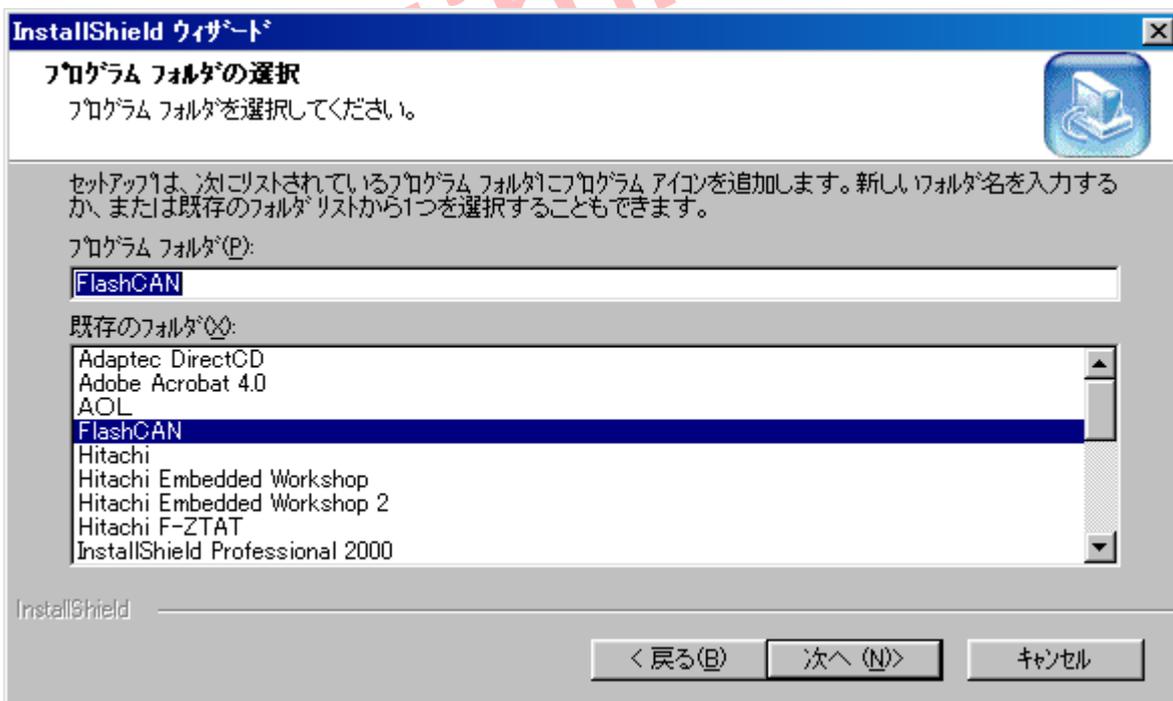
(インストール先を変更する場合は、参照ボタンを押下し、インストール先を選択します。)



手順(5) : 次へボタンを押下します。

(戻るボタンを押下すると手順(3)に戻ります。)

(キャンセルボタンを押下するとインストールの中止確認ダイアログを表示します。)



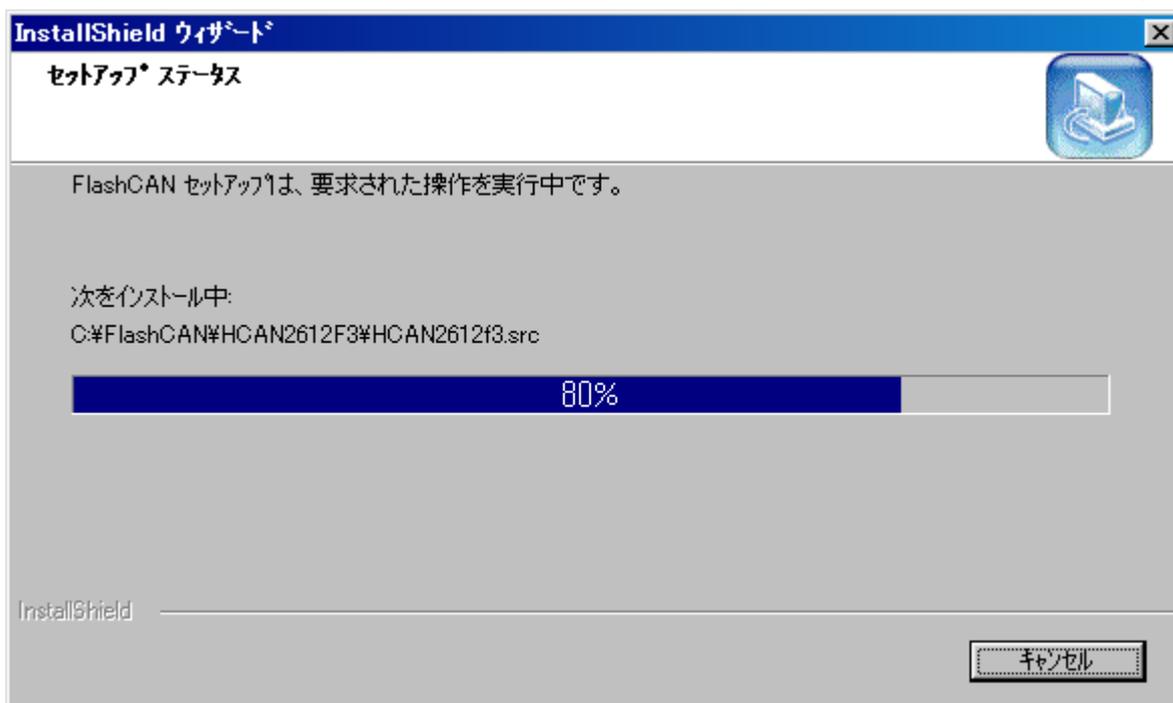
手順(6) : プログラムフォルダを選択し、次へボタンを押下します。

(戻るボタンを押下すると手順(3)に戻ります。)

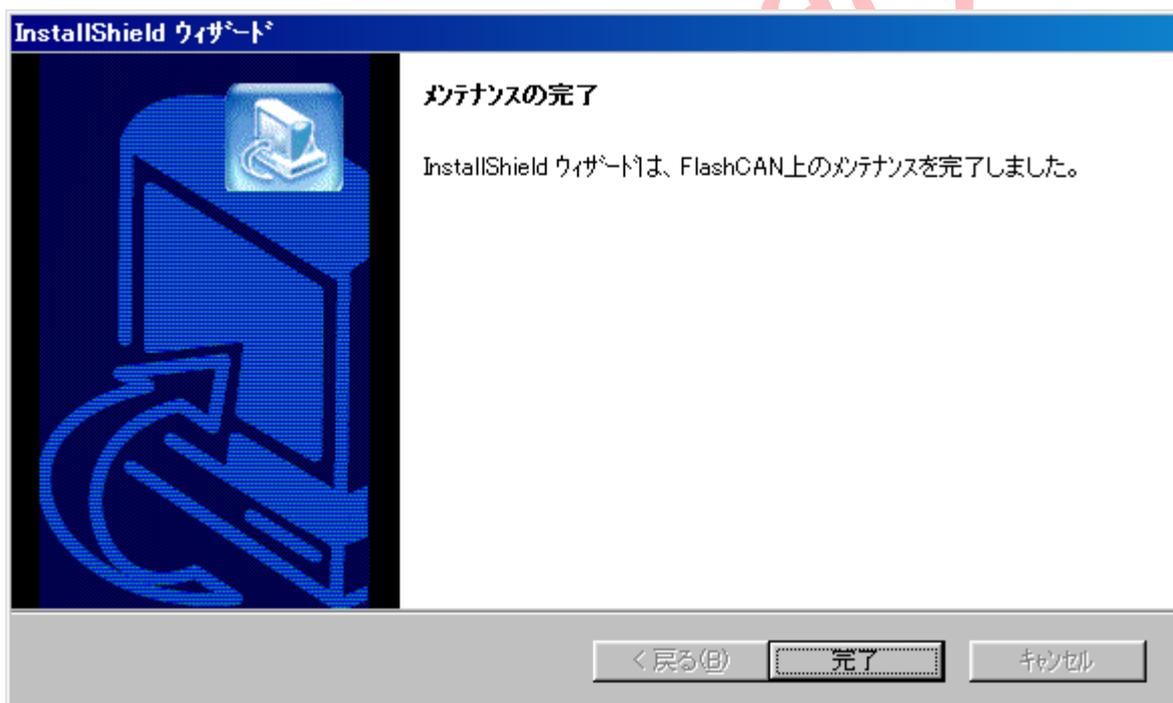
(キャンセルボタンを押下するとインストールの中止確認ダイアログを表示します。)

手順(7) : インストール状況 を表示します。

(キャンセルボタンを押下するとインストールの中止確認ダイアログを表示します。)



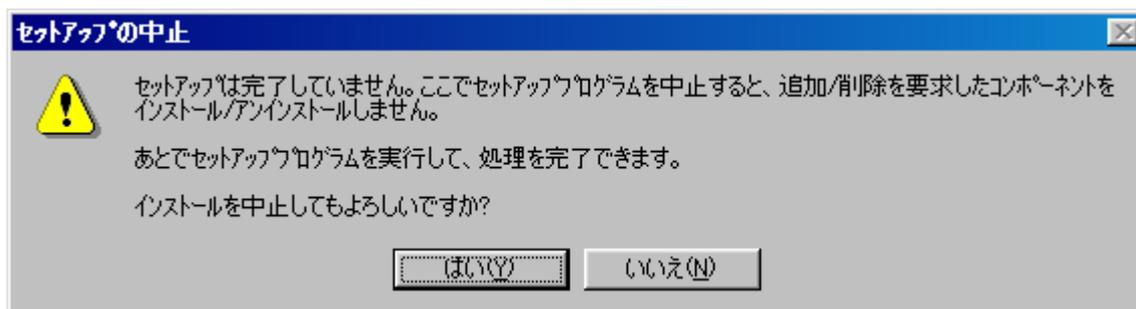
手順(8) : インストールの完了を表示します。完了ボタンを押下します。



以上で、オンボード書き込みツールのインストールが完了します。

[スタートメニュー]-[プログラム]-[FlashCAN]-FlashCAN(ショートカット)を押下し、オンボード書き込みツールを起動します。

・インストールの中止確認ダイアログ



はいボタンを押下すると、インストールを中止します。

いいえボタンを押下すると、インストールを続行します。

・アンインストール

インストール済みの状態で Setup.exe を起動するとアンインストールをします。



OK ボタンを押下し、ウィザードに従いアンインストールをします。

キャンセルボタンを押下するとアンインストールを中止します。

Preliminary

## 8.2 ターゲットボードへのアプリケーション（サンプル）プログラムの初回書き込み

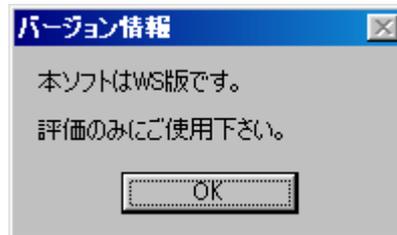
手順(1)：シリアルケーブルをパソコンとターゲットボード間で接続します。

（このアプリケーションノートでは、COM1 に接続します。）

手順(2)：ターゲットボードの電源を ON にします。

手順(3)：ターゲットボードのモードをブートモードに設定します。

手順(4)：FlashCAN.exe を起動します。



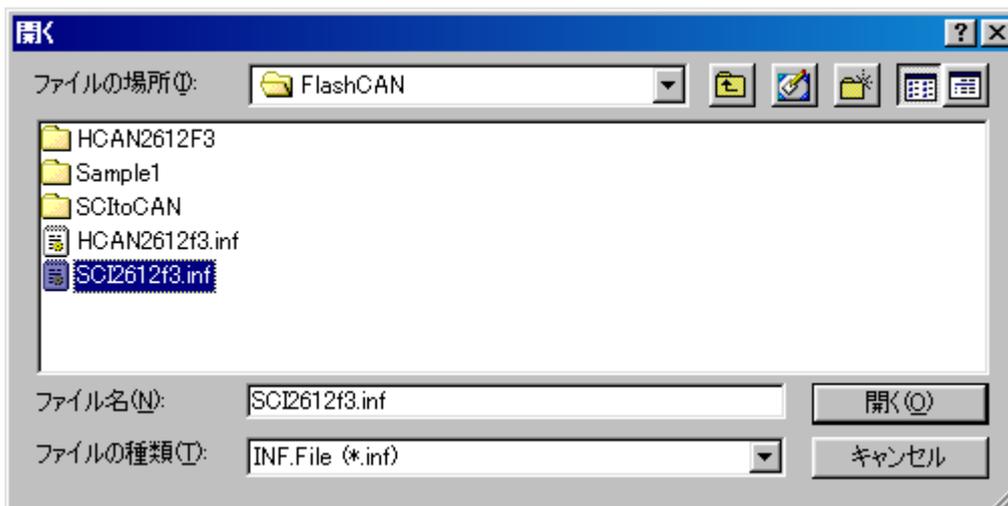
バージョン情報（WS版）を表示します。

手順(5)：OK ボタンを押下するとメインウィンドウが表示されます。



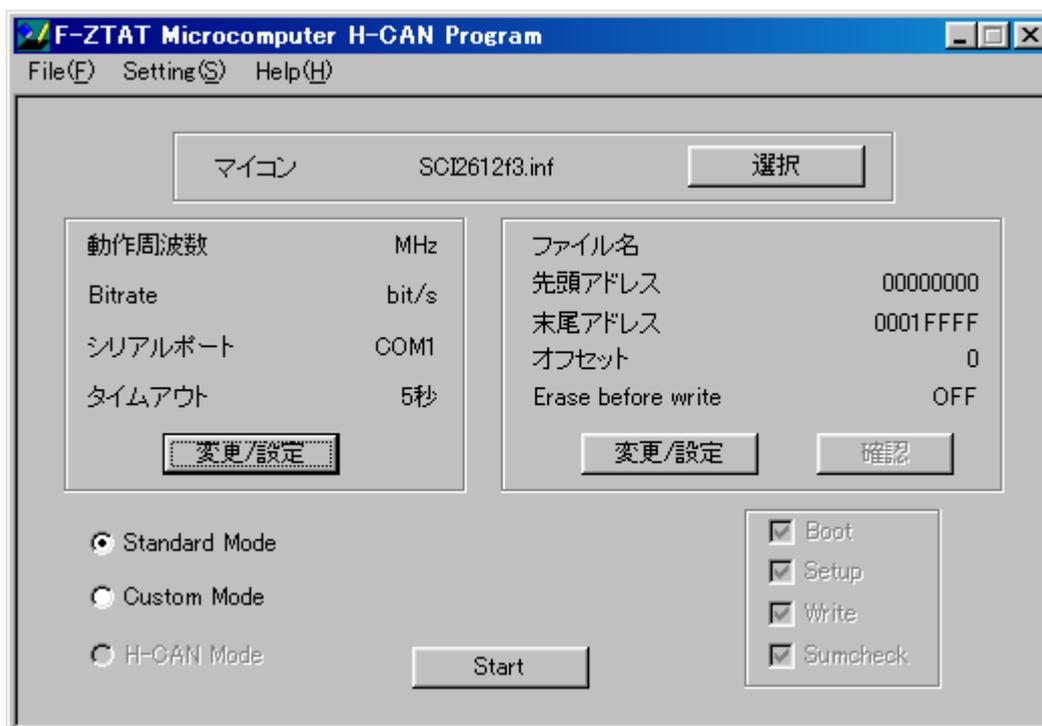
手順(6)：マイコンの選択ボタンを押下し、ファイル選択ダイアログでマイコンを選択します。

手順(7) : 情報ファイル **SCI2612F3.inf** を選択します。

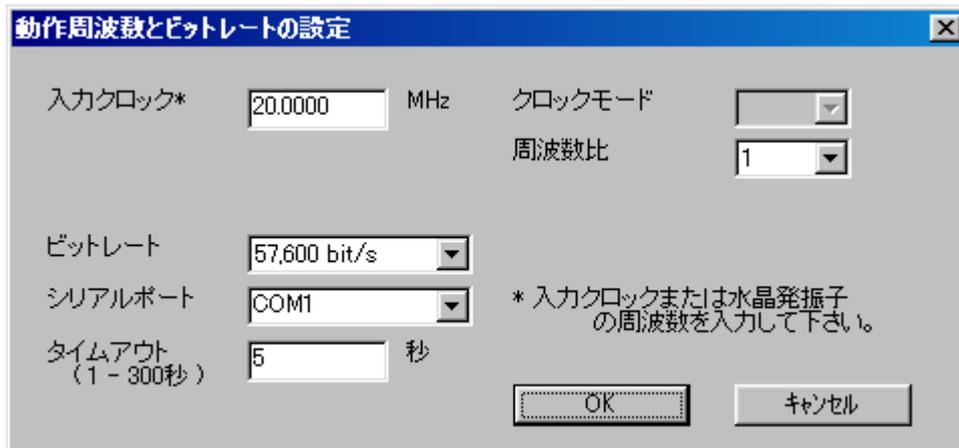


手順(8) : 開くボタンを押下するとメインウィンドウに戻ります。

(キャンセルボタンを押下するとファイルを選択しないでメインウィンドウに戻ります。)



手順(9) : 動作周波数とビットレートの設定の変更/設定ボタンを押下し、動作周波数とビットレートの設定をします。



手順(10) : 入力クロック **20.0000MHz** を設定します。

設定範囲は、SCI2612F3.inf に記載されている 4~20MHz となります。

手順(11) : 周波数比 **1** を選択します。

選択範囲は、SCI2612F3.inf に記載されている 1, 2, 4 となります。

手順(12) : シリアル通信のビットレート **57600bit/s** を選択します。

選択範囲は、2400, 4800, 9600, 19200, 38400, 57600, 115200, (None)となります。

手順(13) : シリアルポート **COM1** を選択します。

選択範囲は、COM1, COM2 となります。

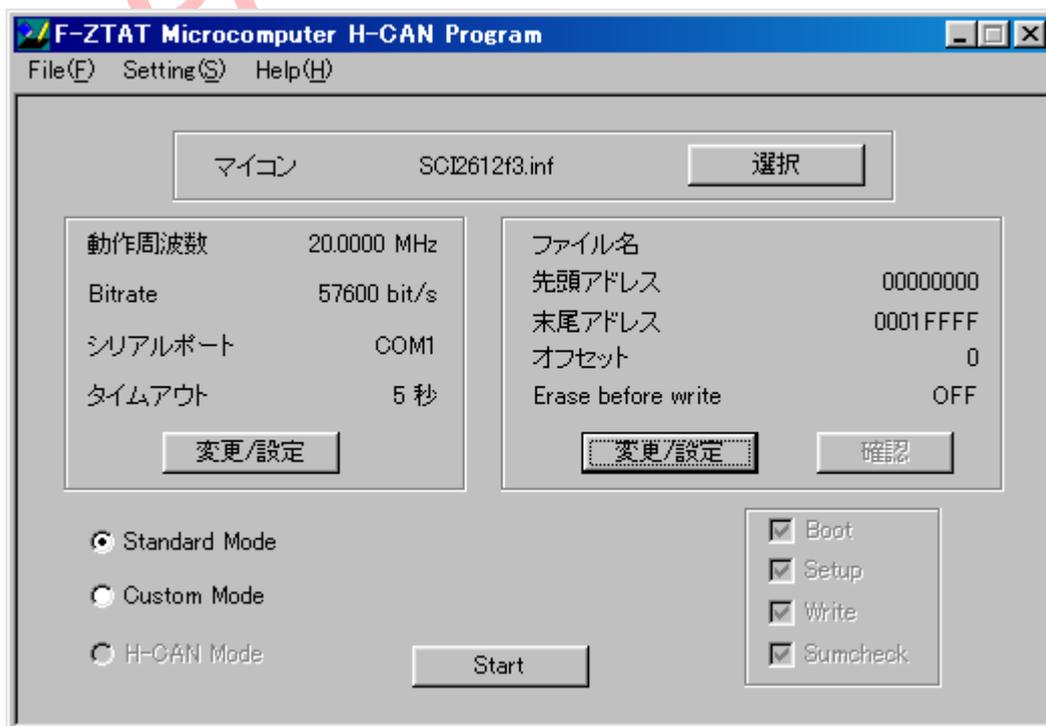
直接キー入力することで、その他のポートを設定することも可能です。

手順(14) : タイムアウト **5** を設定します。

設定範囲は、1~300 となります。

手順(15) : OK ボタンを押下するとメインウィンドウへ戻ります。

(キャンセルボタンを押下すると設定を変更せずにメインウィンドウへ戻ります。)



手順(16) : 書き込みと消去の設定の変更/設定ボタンを押下し、書き込みと消去の設定をします。



手順(17) : 書き込みデータファイル **Sample1.mot** を選択します。

検索ボタンを押下するとファイル選択ダイアログで選択できます。

手順(18) : 先頭アドレス **0x00000000** を選択します。

選択範囲は、SCI2612F3.inf に記載されている 000000 , 000400 , 000800 , 000C00 , 001000 , 008000 , 00C000 , 00E000 , 010000 , 018000 となります。

直接キー入力することで、000000 ~ 01FFFE まで設定可能です。

手順(19) : 末尾アドレス **0x0001FFFF** を選択します。

選択範囲は、SCI2612F3.inf に記載されている 0003FF , 0007FF , 000BFF , 000FFF , 007FFF , 00BFFF , 00DFFF , 00FFFF , 017FFF , 018FFF となります。

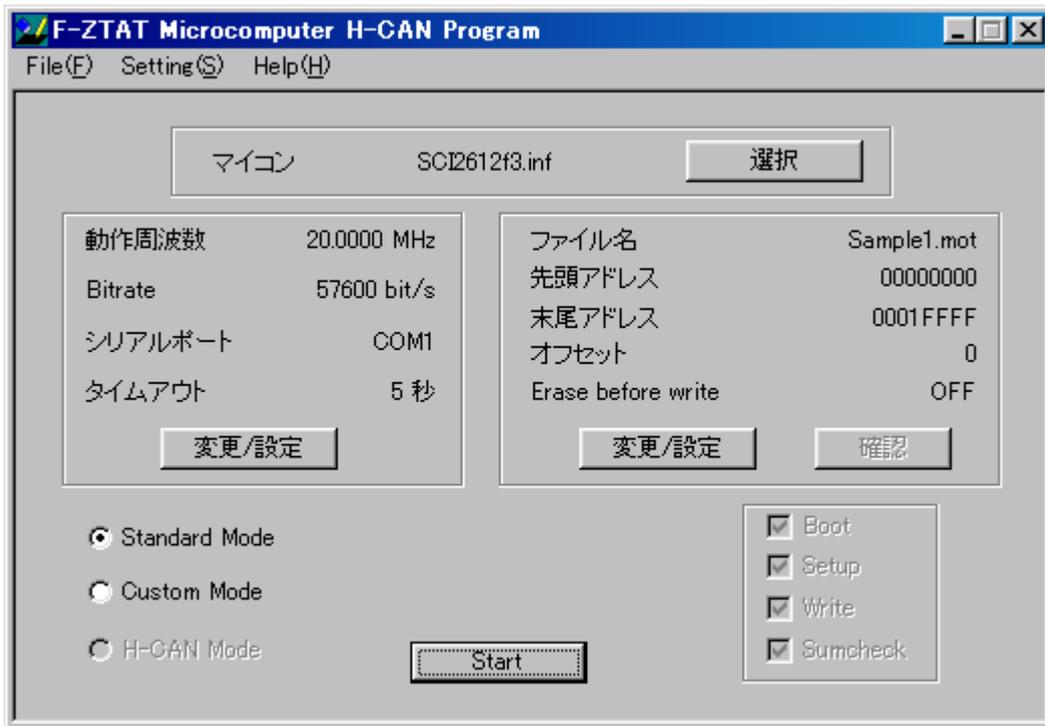
直接キー入力することで、000001 ~ 01FFFF まで設定可能です。

手順(20) : オフセット **0x00000000** を設定します。

設定範囲は、SCI2612F3.inf に記載されている 000000 ~ 01FFFF となります。

手順(21) : OK ボタンを押下するとメインウィンドウへ戻ります。

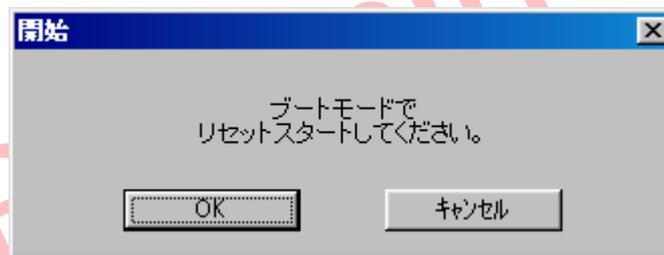
(キャンセルボタンを押下すると設定を変更せずにメインウィンドウへ戻ります。)



手順(22) : 各設定が完了したので、スタートボタンを押下します。

手順(23) : スタートダイアログが表示されます。

ターゲットボードをリセットスタートします。



手順(24) : OK ボタンを押下すると処理の実行を開始します。

(キャンセルボタンを押下するとメインウィンドウへ戻ります。)



手順(25) : ブート処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)



手順(26) : 書き込み処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)

手順(27) : 処理の実行が完了すると、チェックサムが表示されます。



手順(28) : チェックサムを確認し、OK ボタンを押下するとメインウィンドウへ戻ります。

以上で、ターゲットボードへのアプリケーション( サンプル)プログラムの初回書き込みが完了します。  
ターゲットボードのリセットスタートをするとアプリケーション( サンプル)プログラムが起動します。  
パラメータが 0xFF になっているので LED を全点灯表示します。

### 8.3 SCI HCAN 通信変換ボードへの SCI HCAN 通信変換プログラムの書き込み

手順(1) : シリアルケーブルをパソコンと SCI HCAN 通信変換ボード間で接続します。

(このアプリケーションノートでは、COM1 に接続します。)

手順(2) : SCI HCAN 通信変換ボードの電源を ON にします。

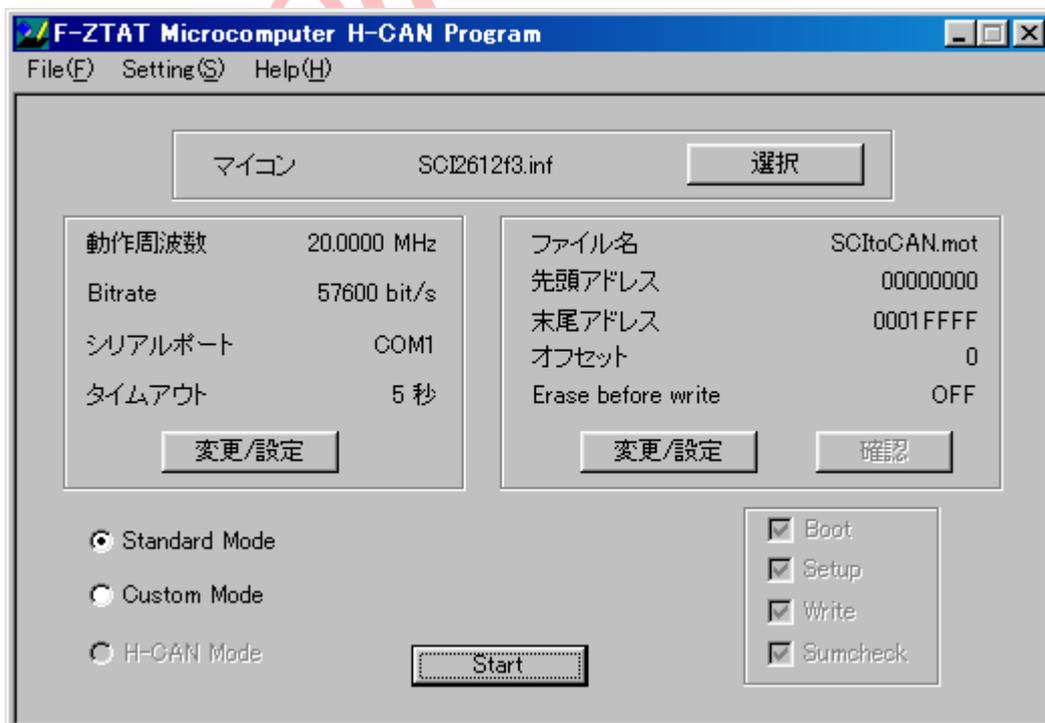
手順(3) : SCI HCAN 通信変換ボードのモードを **ブートモード** に設定します。

手順(4) : ターゲットボードへのアプリケーション (サンプル) プログラムの初回書き込みの  
手順(4) ~ (17) をする。

手順(5) : ターゲットボードへのアプリケーション (サンプル) プログラムの初回書き込みの  
手順(18) で書き込みデータファイル **SCItoCAN.mot** を選択する。



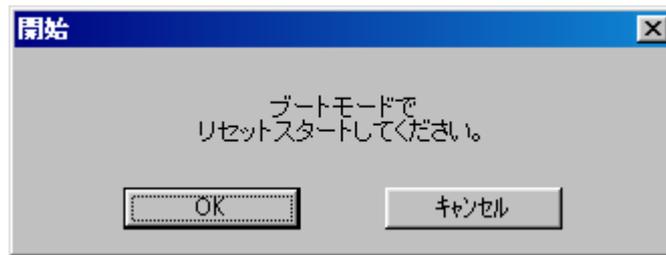
手順(6) : ターゲットボードへのアプリケーション (サンプル) プログラムの初回書き込みの  
手順(19) ~ (21) の書き込みと消去の設定をする。



手順(7) : 各設定が完了したので、スタートボタンを押下します。

手順(8) : スタートダイアログが表示されます。

SCI HCAN 通信変換ボードをリセットスタートします。



手順(9) : OK ボタンを押下すると処理の実行を開始します。

(キャンセルボタンを押下するとメインウィンドウへ戻ります。)



手順(10) : ブート処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)



手順(11) : 書き込み処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)

手順(12) : 処理の実行が完了すると、チェックサムが表示されます。



手順(13) : チェックサムを確認し、OK ボタンを押下するとメインウィンドウへ戻ります。

以上で、SCI HCAN 通信変換ボードへの SCI HCAN 通信変換プログラムの書き込みが完了します。

SCI HCAN 通信変換ボードのリセットスタートをすると SCI HCAN 通信変換プログラムが起動します。

#### 8.4 ユーザプログラムモードでのフラッシュメモリ書き換え

手順(1) : シリアルケーブルをパソコンと SCI HCAN 通信変換ボード間で接続します。

(このアプリケーションノートでは、COM1 に接続します。)

手順(2) : CAN ケーブルをターゲットボードと SCI HCAN 通信変換ボード間で接続します。

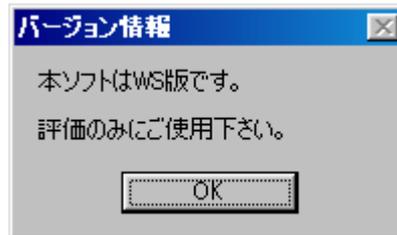
手順(3) : SCI HCAN 通信変換ボードの電源を ON にします。

手順(4) : ターゲットボードの電源を ON にします。

手順(5) : SCI HCAN 通信変換ボードのモードをユーザーモードに設定します。

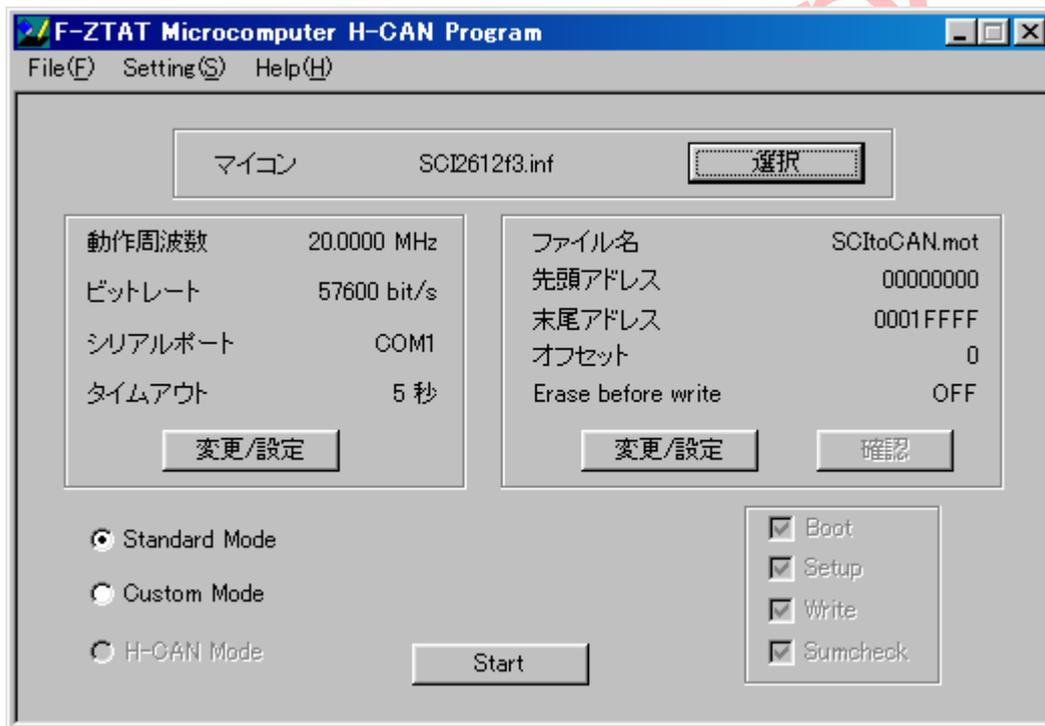
手順(6) : ターゲットボードのモードをユーザーモードに設定します。(リセットスタートします)

手順(7) : FlashCAN.exe を起動します。



バージョン情報 (WS 版) を表示します。

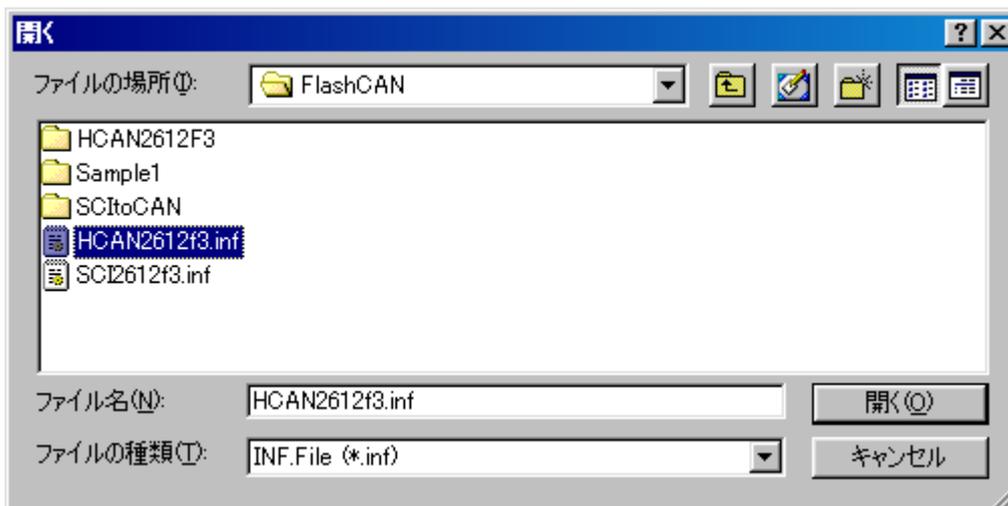
手順(8) : OK ボタンを押下するとメインウィンドウが表示されます。



メインウィンドウには、前回設定した内容が反映されています。

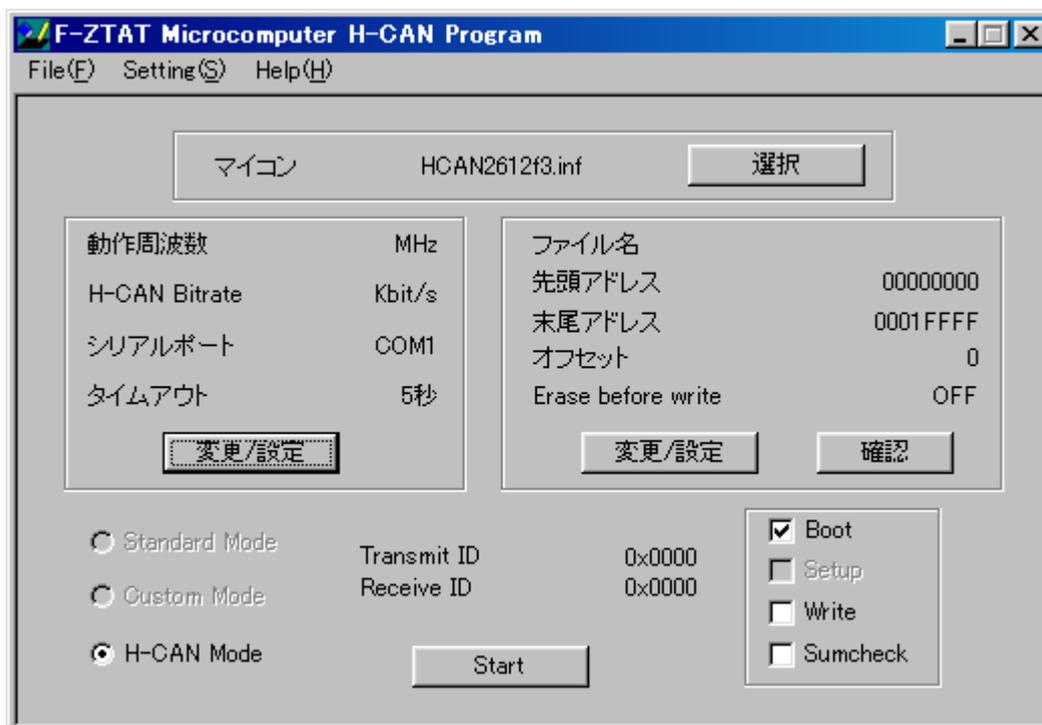
手順(9) : マイコンの選択ボタンを押下し、ファイル選択ダイアログでマイコンを選択します。

手順(10) : 情報ファイル **HCAN2612f3.inf** を選択します。



手順(11) : 開くボタンを押下するとメインウィンドウに戻ります。

(キャンセルボタンを押下するとファイルを選択しないでメインウィンドウに戻ります。)



手順(12) : 動作周波数とビットレートの設定の変更/設定ボタンを押下し、動作周波数とビットレートの設定をします。

手順(13)：ターゲットボードの入力クロック **20.0000MHz** を設定します。

設定範囲は、HCAN2612F3.inf に記載されている 4～20MHz となります。

動作周波数とビットレートの設定					
入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	0000		Synchronization segment	1	time quanta
Receive ID	0000		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta
		OK			キャンセル

手順(14)：送信メールアドレス **ID0x03F9** を設定します。

設定範囲は、0000～07EF となります。

動作周波数とビットレートの設定					
入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	03F9		Synchronization segment	1	time quanta
Receive ID	0000		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta
		OK			キャンセル

手順(15)：受信メールアドレス **ID0x0602** を設定します。

設定範囲は、0000～07EF となります。

動作周波数とビットレートの設定					
入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	03F9		Synchronization segment	1	time quanta
Receive ID	0602		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta
		OK			キャンセル

手順(16) : シリアルポート **COM1** を選択します。

選択範囲は、COM1 , COM2 となります。

直接キー入力することで、その他のポートを設定することも可能です。

入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	03F9		Synchronization segment	1	time quanta
Receive ID	0602		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta

手順(17) : タイムアウト **5** を設定します。

設定範囲は、1 ~ 300 となります。

入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	03F9		Synchronization segment	1	time quanta
Receive ID	0602		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta

手順(18) : Baud rate prescalor **2 system clock** を選択します。

選択範囲は、2 ~ 128 となります。

入力クロック*	20.0000	MHz	Baud rate prescaler	2	system clock
Transmit ID	03F9		Synchronization segment	1	time quanta
Receive ID	0602		Time segment 1	4	time quanta
シリアルポート	COM1		Time segment 2	3	time quanta
タイムアウト (1 - 300秒)	5	秒	Resynchronization jump width	1	time quanta

手順(19) : Synchronization segment 1 time quantum を選択します。

選択範囲は、1 のみとなります。

項目	値	単位/説明
入力クロック*	20,000	MHz
Baud rate prescaler	2	system clock
Transmit ID	03F9	
Receive ID	0602	
シリアルポート	COM1	
タイムアウト (1 - 300秒)	5	秒
Synchronization segment	1	time quanta
Time segment 1	4	time quanta
Time segment 2	3	time quanta
Resynchronization jump width	1	time quanta

手順(20) : Time segment 1 5 time quanta を選択します。

選択範囲は、4 ~ 16 となります。

項目	値	単位/説明
入力クロック*	20,000	MHz
Baud rate prescaler	2	system clock
Transmit ID	03F9	
Receive ID	0602	
シリアルポート	COM1	
タイムアウト (1 - 300秒)	5	秒
Synchronization segment	1	time quanta
Time segment 1	5	time quanta
Time segment 2	3	time quanta
Resynchronization jump width	1	time quanta

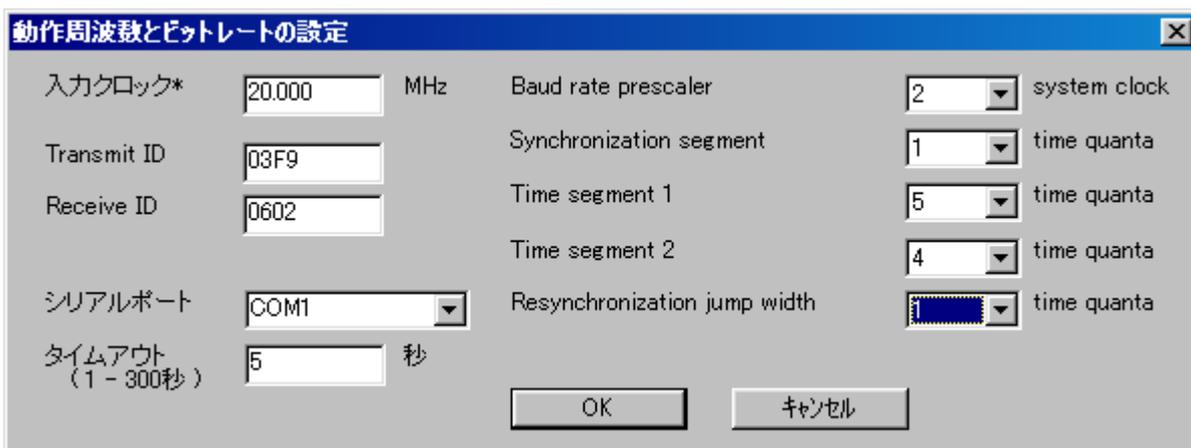
手順(21) : Time segment 2 4 time quanta を選択します。

選択範囲は、3 ~ 8 のみとなります。

項目	値	単位/説明
入力クロック*	20,000	MHz
Baud rate prescaler	2	system clock
Transmit ID	03F9	
Receive ID	0602	
シリアルポート	COM1	
タイムアウト (1 - 300秒)	5	秒
Synchronization segment	1	time quanta
Time segment 1	5	time quanta
Time segment 2	4	time quanta
Resynchronization jump width	1	time quanta

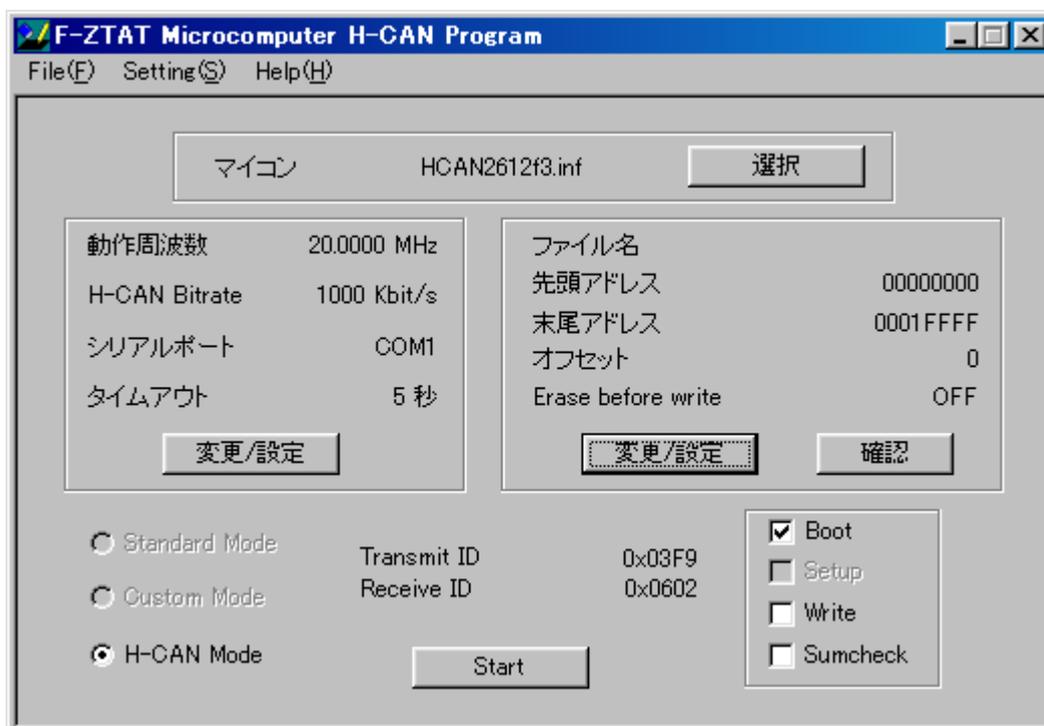
手順(22) : Resynchronization jump width **1 time quanta** を選択します。

選択範囲は、1~4 となります。



手順(23) : OK ボタンを押下するとメインウィンドウへ戻ります。

(キャンセルボタンを押下すると設定を変更せずにメインウィンドウへ戻ります。)



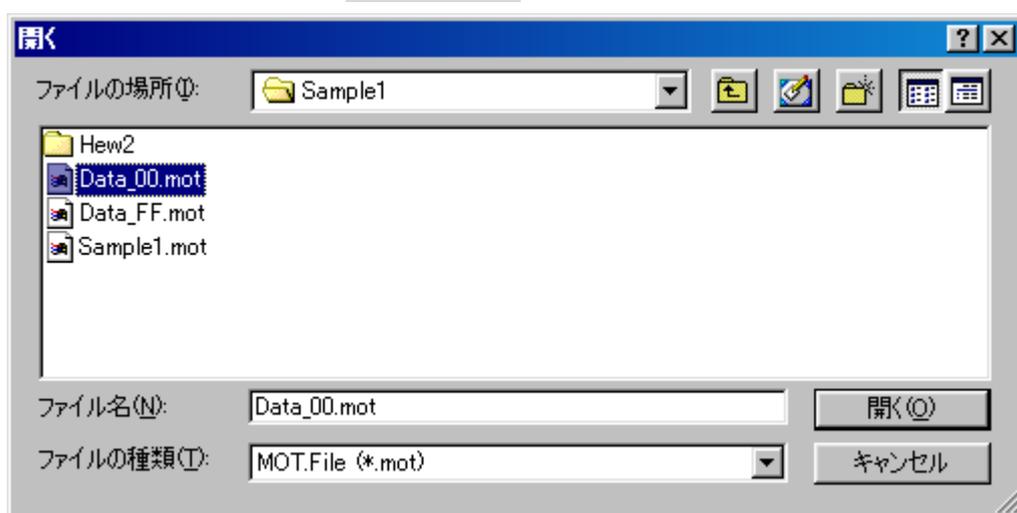
設定した入力クロック, シリアルポート, タイムアウト, 送信メールアドレス ID, 受信メールアドレス ID と設定値を元に計算された H-CAN 通信のビットレートが表示されます。

手順(24) : 書き込みと消去の設定の変更/設定ボタンを押下し、書き込みと消去の設定をします。

手順(25) : 検索ボタンを押下し、書き込みデータファイルを選択します。

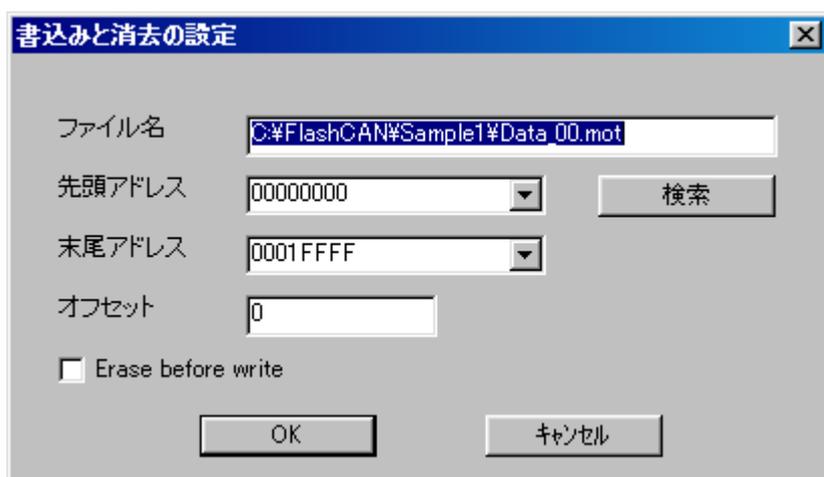


手順(26) : 書き込みデータファイル **Data\_00.mot** を選択します。



手順(27) : 開くボタンを押下すると書き込みと消去の設定ダイアログに戻ります。

(キャンセルボタンを押下するとファイルを選択しないで書き込みと消去の設定ダイアログに戻ります。)



手順(28) : 先頭アドレス **0x00001000** を選択します。

選択範囲は、HCAN2612F3.inf に記載されている 000000 , 000400 , 000800 , 000C00 , 001000 , 008000 , 00C000 , 00E000 , 010000 , 018000 となります。

直接キー入力することで、000000 ~ 01FFFE まで設定可能です。



書き込みと消去の設定

ファイル名 C:\FlashCAN\Sample1\Data\_00.mot

先頭アドレス 00001000 検索

末尾アドレス 0001FFFF

オフセット 0

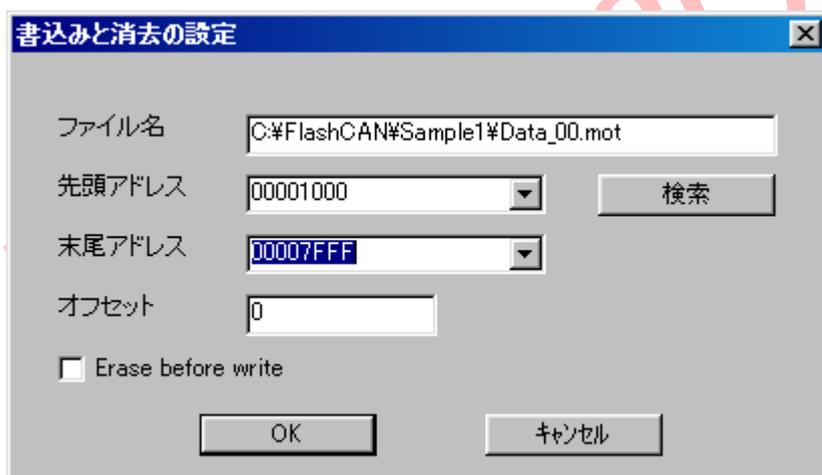
Erase before write

OK キャンセル

手順(29) : 末尾アドレス **0x00007FFF** を選択します。

選択範囲は、HCAN2612F3.inf に記載されている 0003FF , 0007FF , 000BFF , 000FFF , 007FFF , 00BFFF , 00DFFF , 00FFFF , 017FFF , 018FFF となります。

直接キー入力することで、000001 ~ 01FFFF まで設定可能です。



書き込みと消去の設定

ファイル名 C:\FlashCAN\Sample1\Data\_00.mot

先頭アドレス 00001000 検索

末尾アドレス 00007FFF

オフセット 0

Erase before write

OK キャンセル

手順(30) : オフセット **0x00000000** を設定します。

設定範囲は、HCAN2612F3.inf に記載されている **000000 ~ 01FFFF** となります。

書き込みと消去の設定

ファイル名: C:\FlashCAN\Sample1\Data\_00.mot

先頭アドレス: 00001000 [検索]

末尾アドレス: 00007FFF

オフセット: 0

Erase before write

[OK] [キャンセル]

手順(31) : Erase before write **チェックあり(ON)** を選択します。

チェックあり(ON)すると write 処理時、書き込み処理の前に消去処理をします。

チェックなし(OFF)すると write 処理時、消去処理をせずに書き込み処理をします。

書き込みと消去の設定

ファイル名: C:\FlashCAN\Sample1\Data\_00.mot

先頭アドレス: 00001000 [検索]

末尾アドレス: 00007FFF

オフセット: 0

Erase before write

[OK] [キャンセル]

手順(32) : OK ボタンを押下し、消去ブロックの指定ダイアログで消去ブロックを指定します。

消去ブロックの指定

消去対象ブロック: 00001000 - 00007FFF

消去対象ブロックの追加と削除: 00001000 - 00007FFF

[追加]

[削除]

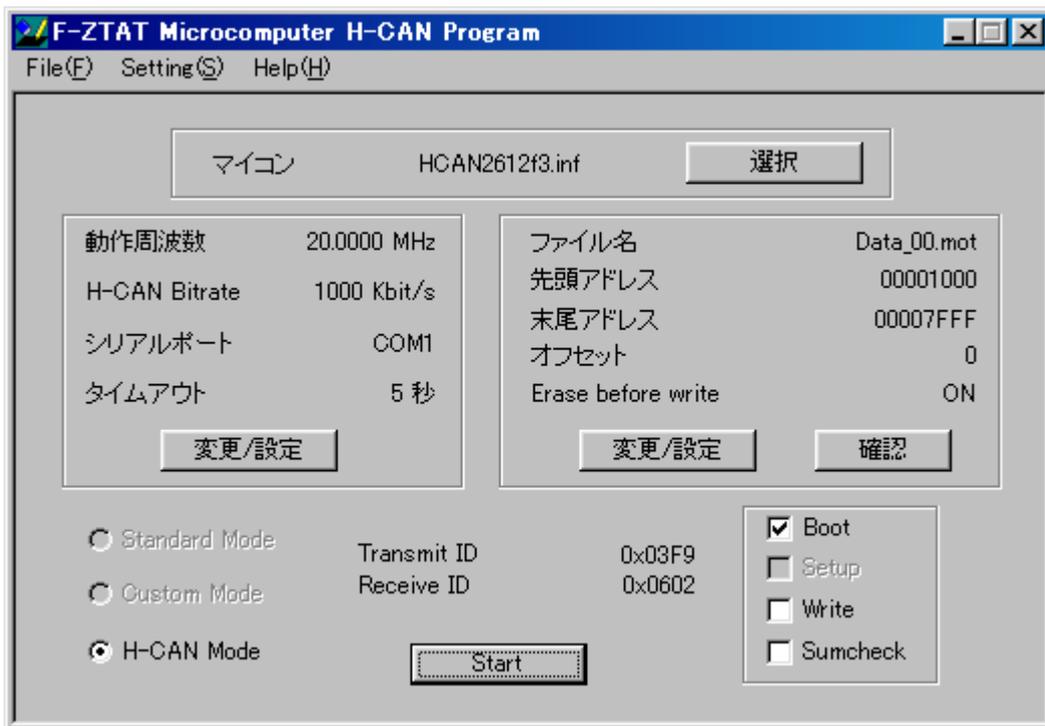
[すべて削除]

[OK] [キャンセル]

選択した先頭アドレス～末尾アドレスの範囲が消去対象ブロックに設定されています。

手順(33) : OK ボタンを押下するとメインウィンドウへ戻ります。

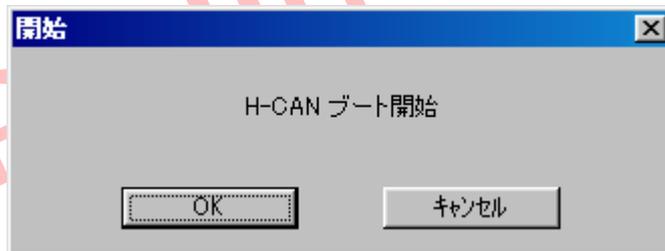
(キャンセルボタンを押下すると書き込みと消去の設定ダイアログへ戻ります。)



手順(34) : 各設定が完了したので、スタートボタンを押下します。

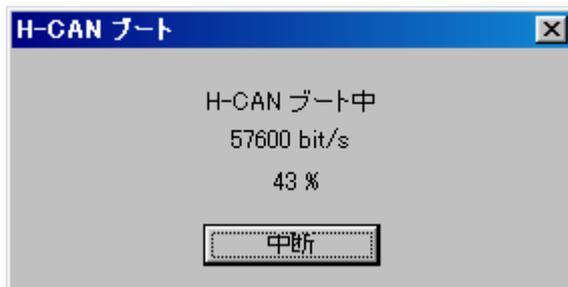
手順(35) : スタートダイアログが表示されます。

SCI HCAN 通信変換ボードをリセットスタートします。



手順(35) : OK ボタンを押下すると処理の実行を開始します。

(キャンセルボタンを押下するとメインウィンドウへ戻ります。)



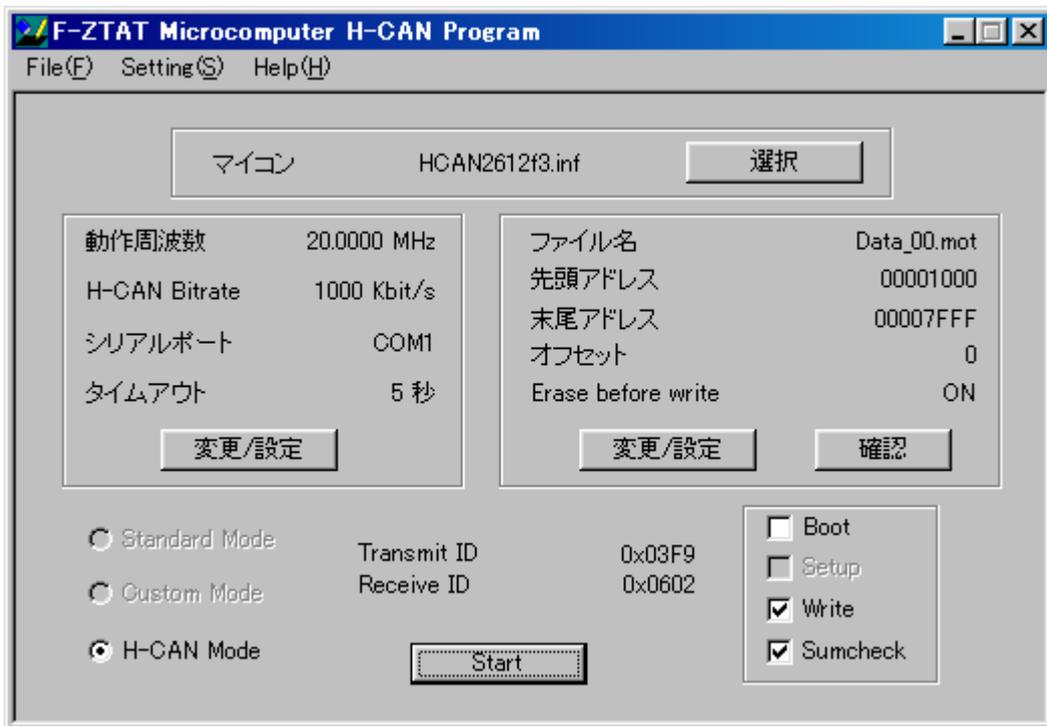
手順(36) : ブート処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)

手順(37) : 処理の実行が完了すると、完了メッセージが表示されます。



手順(38) : OK ボタンを押下するとメインウィンドウへ戻ります。



手順(39) : 実行処理 **Boot** をチェックなし(OFF)にします。

実行処理 **Writ** をチェックあり(ON)にします。

実行処理 **Sumcheck** をチェックあり(ON)にします。

手順(40) : ターゲットボードのモードをユーザプログラムモードに設定します。

手順(41) : 書き込みの各設定が完了したので、スタートボタンを押下します。

手順(42) : スタートダイアログが表示されます。



手順(43) : OK ボタンを押下すると処理の実行を開始します。

(キャンセルボタンを押下するとメインウィンドウへ戻ります。)



手順(44) : 書き込み処理中の表示と進行状況を表示します。

(中断ボタンを押下すると実行中の処理を中断し、メインウィンドウへ戻ります。)

手順(45) : 処理の実行が完了すると、チェックサムが表示されます。



手順(46) : チェックサムを確認し、OK ボタンを押下するとメインウィンドウへ戻ります。

以上で、ユーザプログラムモードでのフラッシュメモリ書き換えが完了します。

ターゲットボードのリセットスタートをするとアプリケーション( サンプル)プログラムが起動します。

パラメータが 0x00 になっているので LED を全点滅表示します。

## 8.5 FlashCAN.exe のエラーメッセージ ( HCAN 対応の追加分 )

### ( 1 ) エラーダイアログボックス

エラーが発生するとダイアログボックスを表示します。

メッセージを確認したら OK ボタンを押下してください。



### ( 2 ) エラーメッセージ一覧

319 【説明】	情報ファイル:H-CAN の指定に誤りがあります 情報ファイル(INF ファイル)の H-CAN の指定に誤りがあります。
650 【説明】	Transmit-ID の指定が正しくありません Transmit-ID(送信メールアドレスID)の指定が不正です。
651 【説明】	Receive-ID の指定が正しくありません Receive-ID(受信メールアドレスID)の指定が不正です。
652 【説明】	Transmit-ID / Receive-ID の指定が正しくありません Transmit-ID(送信メールアドレスID)と Receive-ID(受信メールアドレスID)の指定が同一です。
653 【説明】	TSEG2 の指定が正しくありません TSEG2(Time segment2)の指定が不正です。
654 【説明】	TSEG1 の指定が正しくありません TSEG1(Time segment1)の指定が不正です。
800 【説明】	H-CAN セットアップエラー H コマンドの通信中にエラーを検出しました。 ホスト側が NAK(0x07)を受信した為、失敗しました。
801 【説明】	H-CAN セットアップエラー(確認エラー) H コマンドの通信中にエラーを検出しました。 ホスト側が ACK(0x06)、NAK(0x07)以外を受信した為、失敗しました。
802 【説明】	H-CAN セットアップエラー(タイムアウト) H コマンドの通信中にエラーを検出しました。 ホスト側が受信できずにタイムアウトした為、失敗しました。
803 【説明】	H-CAN 周波数送信エラー H-CAN 周波数送信中にエラーを検出しました。 ホスト側が NAK(0x07)を受信した為、失敗しました。
804 【説明】	H-CAN 周波数送信エラー(確認エラー) H-CAN 周波数送信中にエラーを検出しました。 ホスト側が ACK(0x06)、NAK(0x07)以外を受信した為、失敗しました。
805 【説明】	H-CAN 周波数送信エラー(タイムアウト) H-CAN 周波数送信中にエラーを検出しました。 ホスト側が受信できずにタイムアウトした為、失敗しました。

## 9. 補足説明

### 9.1 ユーザプログラムモードでフラッシュメモリを書き換えるために必要な項目

ユーザプログラムモードでフラッシュメモリを書き換えるには、下記的手段をユーザが準備する必要があります。このアプリケーションノートで使用する手段は、表中の網掛けで示しています。

#### (1) ユーザボードに必要な項目

	ユーザが準備する必要のある項目	手段の例
1	<b>ブートモードで書き込む手段</b> ユーザモード ブートモードを切り換えます。 S C I _ 2 で書き込みデータを供給します。	モード切替えスイッチ、 S C I _ 2 コネクタ
2	<b>FWE端子をハードウェアで切り換える手段</b> ユーザモード ユーザプログラムモードを切り換えます。	モード切替えスイッチ

[注意] FWE端子に常時Highレベルを印加しないでください。また、FWE端子は、CPUがフラッシュメモリをアクセスしていない状態で切り換えてください。

#### (2) アプリケーションに組み込む必要のある項目

	ユーザが準備する必要のある項目	手段の例
1	<b>フラッシュメモリの書き換え処理へ遷移する手段</b> フラッシュメモリの書き換え処理を開始するトリガを受け付けて、書き換え処理へジャンプします。	FWE端子のレベルセンス、 外部割込み、 SCIによるコマンド受信、 HCANによるコマンド受信、 など
2	<b>書き込み/消去制御プログラムをRAMへ転送する手段</b> フラッシュメモリの書き込み/消去はRAM上のプログラムで制御する必要があります。このため書き込み/消去制御プログラムをRAMへ転送し、転送後のプログラムへジャンプします。	ROMからRAMへ転送、 SCIで外部から転送、 HCANで外部から転送、 など

#### (3) ホストに必要な項目

	ユーザが準備する必要のある項目	手段の例
1	<b>書き込み/消去制御プログラム</b> モトローラ形式のロードモジュールです。 書き込み/消去のアルゴリズムは、ハードウェアマニュアルに沿ったものにしてください。また、消去ブロックの指示/応答や書き込みデータを受信するためにホストに合わせた送受信の機能を組み込む必要があります。	FDT.exeに付属、 FlashCAN.exeに付属、 ユーザ作成、 など
2	<b>書き込みデータを供給する手段(ホスト)</b> 上記の(2) 1, 2のシーケンス制御及び、書き込みデータを転送するためにホストが必要です。	FDT.exe、 FlashCAN.exe、 ユーザ作成のツール、 など

## 9.2 ユーザプログラムモードとブートモードとの違い

オンボードでフラッシュメモリを書き換えるモードは、ユーザプログラムモードとブートモードの2通りがあります。それぞれのモードでの違いを示します。

項目	ユーザプログラムモード	ブートモード
アプリケーションプログラムの実行	フラッシュメモリにダウンロードされているアプリケーションプログラムが実行されます。 このアプリケーションプログラムにフラッシュメモリの書き換え処理を組み込んでおくことにより、ユーザプログラムモードでのフラッシュメモリの書き換えを実現します。	フラッシュメモリにダウンロードされているアプリケーションプログラムは実行されません。 F-ZTATマイコンに内蔵されているブートプログラムが実行されます。
フラッシュメモリの書き換えで使用するインタフェース	ユーザシステムに合わせて、使用するインタフェースを自由に選択できます。 例えば、SCI、HCANなどが使用できます。	SCIを使用します。 通信速度は、F-ZTATマイコンが自動的に合せ込みます。 また、プロトコルは固定です。
書き込み / 消去制御プログラムの転送先	RAMの全エリアH'FFE000 ~ H'FFEFBF(4032バイト)を使用できます。	RAMのH'FFE800 ~ H'FFEFBF番地(1984バイト)へ転送します。
消去ブロック	消去するブロックを自由に指定できます。 フラッシュメモリの消去を制御するプログラム(消去制御プログラム)は、ユーザが準備し、RAMへ転送する必要があります。	F-ZTATマイコンが自動的に全ブロックを消去します。
書き込み	消去ブロック単位での書き換えが可能です。消去したブロックに対してのみ書き込みます。 フラッシュメモリへの書き込みを制御するプログラム(書き込み制御プログラム)は、ユーザが準備し、RAMへ転送する必要があります。	全ブロックが消去されるので、全面的に書き込む必要があります。 フラッシュメモリへの書き込みを制御するプログラム(書き込み制御プログラム)は、ユーザが準備し、RAMへ転送する必要があります。
モードへの遷移方法	2通りの方法があります。 MD0=1, MD1=1, MD2=1, FWE=1でリセット。 MD0=1, MD1=1, MD2=1, FWE=0ユーザモードで実行中に、FWE=1に切り換える。 このため、ユーザシステムをリセットせずにフラッシュメモリの書き換えることも可能です。	MD0=1, MD1=1, MD2=0, FWE=1でリセット。

### 9.3 Eビット、Pビットの印加時間の実測方法

フラッシュメモリの消去、書き込みは、FLMCRレジスタのEビット、Pビットをセットして電圧を印加することで実現しています。電圧の印加時間はハードウェアマニュアルに記載されています。

Eビット、Pビットの印加時間を誤って短くすると消去/書き込みができません。また、誤って長くしてしまうと過剰消去/過剰書き込みとなりデバイスの永久破壊につながります。Eビット、Pビットをセットするときは、プログラムの暴走などに備えてあらかじめ内蔵ウォッチドックタイマを設定してください。

このアプリケーションノートのサンプルプログラムでは、Eビット、Pビットの印加時間をソフトウェアループのループ回数を調整して実現しています。そのため、動作周波数に合わせてループ回数を増減する必要があります。ソフトウェアループにかかる時間は机上の計算やシミュレータデバッガにより求められますが、Eビット、Pビットの印加時間は非常に重要ですので外部から実測して確認することをお奨めします。

Eビット、Pビットの実測は、内蔵I/OポートへEビット、PビットのHIGH/LOWを出力することにより行ないます。Eビット、PビットをHIGH/LOWする同タイミングで内蔵I/Oポートへも出力してください。この内蔵I/Oポートの出力信号をオシロスコープ、ロジックアナライザなどで外部から時間計測します。

ソース例 (PビットのOn/Offを内蔵I/OポートのP0<sub>0</sub>へ出力する)

```
;===== WRITE パルス印加 =====
      BSET.B  #0,@PORT0      ; Pビットの実測のため、ポート0の0ビットをセット
      BSET.B  #P,@ER6       ; Pビットセット(書き込み)
FWRT40 DEC.L  #1,ER3        ; 書き込み時間: 10µS or 30µS or 200µS
      BNE    FWRT40:16
;=====
      MOV.W   @WLOOP5,E0
      BCLR.B  #P,@ER6       ; Pビットをクリア
      BCLR.B  #0,@PORT0    ; Pビットの実測のため、ポート0の0ビットをクリア
```

# H8S/2612F 内蔵 CAN によるフラッシュメモリの書換え アプリケーションノート



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