

**TOSHIBA**

東芝 オリジナル CMOS 16ビット マイクロコントローラ

**TLCS-900/L1 シリーズ**

**TMP91FW40FG**

Not Recommended  
for New Design

株式会社 **東芝** セミコンダクター社

## はじめに

この度は弊社 16 ビットマイクロコントローラ TLCS-900/L1 シリーズ、TMP91FW40 をご利用いただき、誠にありがとうございます。

本 LSI をご利用になる前に、「使用上の注意、制限事項」の章を参照されますことをお願いいたします。

Not Recommended  
for New Design

低電圧/低消費電力

## CMOS 16 ビット マイクロコントローラ TMP91FW40FG

### 1. 概要と特長

TMP91FW40 は、低電圧/低消費電力動作が可能な高速・高機能 16 ビットマイクロコントローラです。

TMP91FW40FG は、100 ピンフラットパッケージ製品です。特長は次のとおりです。

- (1) オリジナル 16 ビット CPU (900/L1\_CPU 使用)
  - TLCS-90/900 と命令モニタで上位互換
  - 16 M バイトのリニアアドレス空間
  - 汎用レジスタ&レジスタバンク方式
  - 16 ビット乗除算命令、ビット転送/演算命令
  - マイクロ DMA: 4 チャンネル (593 ns /2 バイト@27 MHz)
- (2) 最小命令実行時間: 148 ns (@27 MHz)
- (3) 内蔵 RAM: 4 K バイト
- (4) 内蔵 ROM: 128 K バイトフラッシュメモリ  
: 4K バイトマスク ROM (ブート機能用)

### 当社半導体製品取り扱い上のごお願い

20070701-JA GENERAL

- 当社は品質、信頼性の向上に努めておりますが、一般に半導体製品は誤作動したり故障することがあります。当社半導体製品をご使用いただく場合は、半導体製品の誤作動や故障により、生命・身体・財産が侵害されることのないように、購入者側の責任において、機器の安全設計を行うことをお願いします。なお、設計に際しては、最新の製品仕様をご確認の上、製品保証範囲内でご使用いただくと共に、考慮されるべき注意事項や条件について「東芝半導体製品の取り扱い上のご注意とお願い」、「半導体信頼性ハンドブック」などでご確認ください。
- 本資料に掲載されている製品は、一般的電子機器（コンピュータ、パーソナル機器、事務機器、計測機器、産業用ロボット、家電機器など）に使用されることを意図しています。特別に高い品質・信頼性が要求され、その故障や誤作動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機器（原子力制御機器、航空宇宙機器、輸送機器、交通信号機器、燃焼制御、医療機器、各種安全装置など）にこれらの製品を使用すること（以下“特定用途”という）は意図もされていませんし、また保証もされていません。本資料に掲載されている製品を当該特定用途に使用することは、お客様の責任でなされることとなります。
- 本資料に掲載されている製品を、国内外の法令、規則及び命令により製造、使用、販売を禁止されている応用製品に使用することはできません。
- 本資料に掲載してある技術情報は、製品の代表的動作・応用を説明するためのもので、その使用に際して当社及び第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。
- 本資料に掲載されている製品の RoHS 適合性など、詳細につきましては製品個別に必ず弊社営業窓口までお問合せください。本資料に掲載されている製品のご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令などの法令を十分調査の上、かかる法令に適合するようご使用ください。お客様が適用される法令を遵守しないことにより生じた損害に関して、当社は一切の責任を負いかねます。
- 本資料の掲載内容は、技術の進歩などにより予告なしに変更されることがあります。

本製品は、米国 SST 社（Silicon Storage Technology, Inc.）からライセンスを受けた Super Flash® 技術を使用しています。Super Flash® は SST 社の登録商標です。

- (5) 8ビットタイマ: 4チャンネル
- (6) 16ビットタイマ: 3チャンネル
- (7) デバイダ出力
- (8) 汎用シリアルインタフェース: 4チャンネル
  - UART/同期両モード対応
- (9) 10ビットADコンバータ(サンプルホールド回路内蔵): 4チャンネル
- (10) ウォッチドッグタイマ
- (11) キーオンウエイクアップ: 4チャンネル
- (12) RTC(リアルタイムクロック)
  - TC8521Aを基本とした仕様
- (13) メロディ/アラームジェネレータ
- (14) プログラム修正機能: 6バンク
- (15) LCDドライバ/コントローラ(降圧タイプ、基準電圧=VCC)
  - LCD直接駆動可能(40~8セグメント×4コモン)
  - 1/4, 1/3, 1/2 デューティ, スタティック駆動の選択
- (16) 割り込み機能: 43本
  - CPU 9本 ..... ソフトウェア割り込み命令、未定義命令実行違反
  - 内部 27本 ..... 7レベルの優先順位の設定が可能
  - 外部 7本 ..... 7レベルの優先順位の設定が可能  
(2本はエッジの極性選択可能)
- (17) 入出力ポート: 61端子
- (18) スタンバイ機能  
3種類のHALTモード(プログラマブルIDLE2, IDLE1, STOP)
- (19) クロック制御機能
  - 低周波クロック ( $f_s = 32.768 \text{ kHz}$ )
- (20) 動作電圧
  - Flash読み出し動作時  
 $V_{CC} = 2.7 \sim 3.6 \text{ V}$  ( $f_c \text{ max} = 27 \text{ MHz}$ )  
 $V_{CC} = 2.2 \sim 3.6 \text{ V}$  ( $f_c \text{ max} = 16 \text{ MHz}$ )
  - Flash書き込み/消去動作時  
 $V_{CC} = 2.7 \sim 3.6 \text{ V}$  ( $f_c \text{ max} = 27 \text{ MHz}$ )
- (21) パッケージ: LQFP100-P-1414-0.50F



## 2. ピン配置とピン機能

### 2.1 ピン配置図

TMP91FW40 のピン配置図および入出力ピンの名称と概略機能を示します。  
 TMP91FW40FGピン配置図は、図 2.1.1のとおりです。

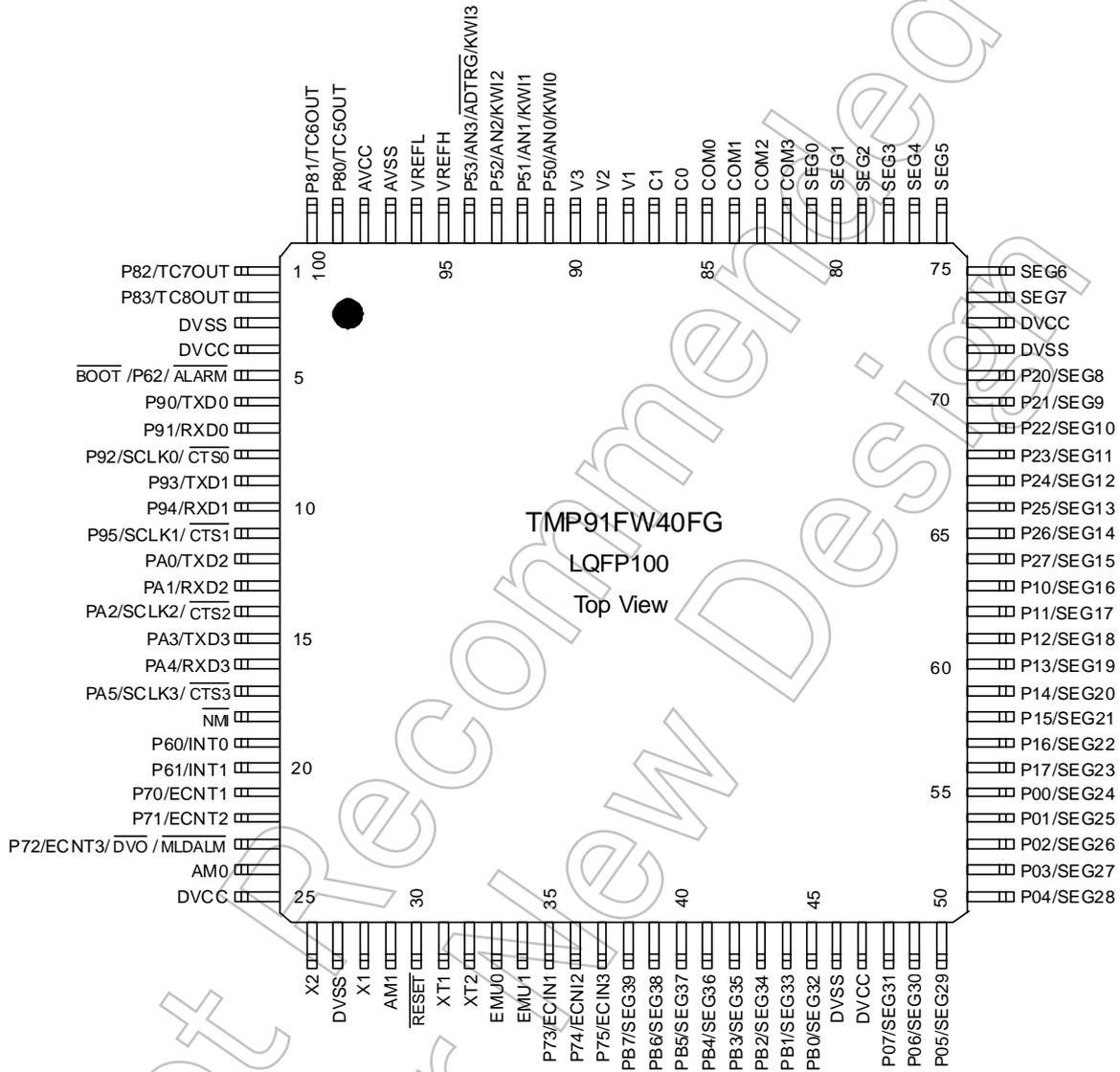


図 2.1.1 ピン配置図 (100 ピン LQFP Top view)

## 2.2 ピン名称と機能

入出力ピンの名称と機能は、表 2.2.1~ 表 2.2.2のとおりです。

表 2.2.1 ピン名称と機能 (1/2)

ピン名称	ピン数	入出力	機能
P50~P53 AN0~AN3 ADTRG KW10~KW13	4	入力 入力 入力 入力	ポート 5: 入力専用ポートです。 アナログ入力: AD コンバータの入力です。 AD トリガ: AD コンバータの外部スタート要求端子です。(P53 と兼用です) キーオンウェイクアップ入力(P50~P53 と兼用です)。
P60 INT0	1	入力 入力	ポート 60: 入力専用ポートです。 割り込み要求端子 0: プログラマブル割り込み (レベル/立ち上がり/立ち下がりエッジ) 要求端子です。
P61 INT1	1	入出力 入力	ポート 61: 入出力ポートです。 割り込み要求端子 1: プログラマブル割り込み (レベル/立ち上がり/立ち下がりエッジ) 要求端子です。
P62 ALARM BOOT	1	入出力 出力 入力	ポート 62: 入出力ポートです。 RTC アラーム出力端子 Flash 用ブートモード制御端子(91FW40 専用、RESET 期間 Pull-Up されます) リセット解除時、Low レベルでシングルブートモードが起動されます。
P70 ECNT1	1	入出力 入力	ポート 70: 入出力ポートです。 16 ビットタイマ 1 入力: 16 ビットタイマ TC1 のカウント制御入力になります。
P71 ECNT2	1	入出力 入力	ポート 71: 入出力ポートです。 16 ビットタイマ 2 入力: 16 ビットタイマ TC2 のカウント制御入力になります。
P72 ECNT3 DVO MLDALM	1	入出力 入力 出力 出力	ポート 72: 入出力ポートです。 16 ビットタイマ 3 入力: 16 ビットタイマ TC3 のカウント制御入力になります。 デバイダ出力端子 メロディ/アラーム出力端子
P73 ECIN1	1	入出力 入力	ポート 73: 入出力ポートです。 16 ビットタイマ 1 入力: 16 ビットタイマ TC1 のカウント入力になります。
P74 ECIN2	1	入出力 入力	ポート 74: 入出力ポートです。 16 ビットタイマ 2 入力: 16 ビットタイマ TC2 のカウント入力になります。
P75 ECIN3	1	入出力 入力	ポート 75: 入出力ポートです。 16 ビットタイマ 3 入力: 16 ビットタイマ TC3 のカウント入力になります。
P80 TC5OUT	1	入出力 出力	ポート 80: 入出力ポートです。(大電流ポート) 8 ビットタイマ 5 出力: 8 ビットタイマ TC5 の出力端子です。 プログラムによりオープンドレイン出力端子となります。
P81 TC6OUT	1	入出力 出力	ポート 81: 入出力ポートです。(大電流ポート) 8 ビットタイマ 6 出力: 8 ビットタイマ TC6 の出力端子です。 プログラムによりオープンドレイン出力端子となります。
P82 TC7OUT	1	入出力 出力	ポート 82: 入出力ポートです。(大電流ポート) 8 ビットタイマ 7 出力: 8 ビットタイマ TC7 の出力端子です。 プログラムによりオープンドレイン出力端子となります。
P83 TC8OUT	1	入出力 出力	ポート 83: 入出力ポートです。(大電流ポート) 8 ビットタイマ 8 出力: 8 ビットタイマ TC8 の出力端子です。 プログラムによりオープンドレイン出力端子となります。
P90 TXD0	1	入出力 出力	ポート 90: 入出力ポートです。 シリアル 0 送信データ プログラムによりオープンドレイン出力端子となります。
P91 RXD0	1	入出力 入力	ポート 91: 入出力ポートです。 シリアル 0 受信データ
P92 SCLK0 CTS0	1	入出力 入出力 入力	ポート 92: 入出力ポートです。 シリアル 0 クロック入出力 シリアル 0 データ送信可能 (Clear to send)

表 2.2.2 ピン名称と機能 (2/2)

ピン名称	ピン数	入出力	機能
P93 TXD1	1	入出力 出力	ポート 93: 入出力ポートです。 シリアル 1 送信データ プログラムによりオープンドレイン出力端子となります。
P94 RXD1	1	入出力 入力	ポート 94: 入出力ポートです。 シリアル 1 受信データ
P95 SCLK1 CTS1	1	入出力 入出力 入力	ポート 95: 入出力ポートです。 シリアル 1 クロック入出力 シリアル 1 データ送信可能 (Clear to send)
PA0 TXD2	1	入出力 出力	ポート A0: 入出力ポートです。 シリアル 2 送信データ プログラムによりオープンドレイン出力端子となります。
PA1 RXD2	1	入出力 入力	ポート A1: 入出力ポートです。 シリアル 2 受信データ
PA2 SCLK2 CTS2	1	入出力 入出力 入力	ポート A2: 入出力ポートです。 シリアル 2 クロック入出力 シリアル 2 データ送信可能 (Clear to send)
PA3 TXD3	1	入出力 出力	ポート A3: 入出力ポートです。 シリアル 3 送信データ プログラムによりオープンドレイン出力端子となります。
PA4 RXD3	1	入出力 入力	ポート A4: 入出力ポートです。 シリアル 3 受信データ
PA5 SCLK3 CTS3	1	入出力 入出力 入力	ポート A5: 入出力ポートです。 シリアル 3 クロック入出力 シリアル 3 データ送信可能 (Clear to send)
SEG0~SEG7	8	出力	セグメント出力
P20~P27 SEG8~SEG15	8	入出力 出力	ポート 2: 入出力ポートです。 セグメント出力
P10~P17 SEG16~SEG23	8	入出力 出力	ポート 1: 入出力ポートです。 セグメント出力
P00~P07 SEG24~SEG31	8	入出力 出力	ポート 0: 入出力ポートです。 セグメント出力
PB0~PB7 SEG32~SEG39	8	入出力 出力	ポート B: 入出力ポートです。 セグメント出力
C0,C1	2		LCD 駆動電源
V1~V3	3		LCD 駆動電源
COM0~COM3	4		コモン出力
NMI	1	入力	ノンマスカブル割り込み要求端子: 立ち下がリエッジの割り込み要求端子です。プログラムにより、立ち上がりエッジでも割り込み要求可能となります (シュミット入力)。
AM0, AM1	2	入力	動作モード: AM1 = "1", AM0 = "1" に固定してください。
EMU0	1	出力	"開放" してください。
EMU1	1	出力	"開放" してください。
RESET	1	入力	リセット: LSI を初期化します (シュミット入力、プルアップ抵抗付き)。
VREFH	1	入力	AD コンバータ用基準電源入力端子です。(H)
VREFL	1	入力	AD コンバータ用基準電源入力端子です。(L)
AVCC	1		AD コンバータ電源端子
AVSS	1		AD コンバータ GND 端子 (0 V)
X1/X2	2	入出力	高速発振子接続端子
XT1/XT2	2	入出力	低速発振子接続端子
DVCC	4		電源端子 (全 DVCC 端子を電源に接続してください。)
DVSS	4		GND 端子 (全 DVSS 端子を GND (0 V) に接続してください。)

### 3. 動作説明

ここでは、TMP91FW40 の機能および基本動作について説明します。  
ここに記載されていない機能については、TMP91CW40 のデータシートを参照してください。

#### 3.1 CPU

TMP91FW40 には、高性能な 16 ビット CPU (900/L1 CPU) が内蔵されています。CPU の動作については、“TLCS-900/L1 CPU” を参照してください。

ここでは、“TLCS-900/L1 CPU” にて説明されていない TMP91FW40 機能について説明します。

##### 3.1.1 リセット動作

本デバイスにリセットをかけるには電源電圧が動作範囲内であり、内部高周波発振器の発振が安定した状態で少なくとも 10 システムクロック間 (27MHz クロック発振時で 1  $\mu$ s)、RESET 入力を “Low” レベルにしてください。また、電源投入時は RESET 入力が “Low” レベルで電源電圧が動作範囲内になり、内部高周波発振器の発振が安定した状態で少なくとも 10 システムクロック間、RESET 入力の “Low” レベルを保持してください。

なお、リセット動作にてシステムクロック  $f_{SYS}$  は  $f_c/2$  となります。

リセットが受け付けられると、CPU は、

- プログラムカウンタ PC をアドレス FFFF00H~FFFF02H に格納されているリセットベクタに従いセット  
PC<7:0> ← アドレス FFFF00H の値  
PC<15:8> ← アドレス FFFF01H の値  
PC<23:16> ← アドレス FFFF02H の値
- スタックポインタ XSP を 100H にセット
- ステータスレジスタ SR の IFF2~IFF0 ビットを “111” にセット (割り込みレベルのマスキングレジスタをレベル 7 にセット)
- ステータスレジスタ SR の MAX ビットを “1” にセット (マキシマムモードにセット)
- ステータスレジスタ SR の RFP2~REPO ビットを “000” にクリア (レジスタバンクを 0 にセット)

を行い、リセットが解除されると、セットされた PC に従い命令の実行を開始します。なお、上記以外の CPU 内部のレジスタは変化しません。

また、リセットが受け付けられると、内蔵 I/O およびポート、その他の端子は下記のとおりとなります。

- 内蔵 I/O のレジスタを初期化
- ポート端子 (内蔵 I/O 用にも使える兼用端子を含む) を、汎用入力ポートまたは汎用出力ポートのモードにセット

注) リセット動作により、CPU の PC, SR, XSP 以外のレジスタ、内蔵 RAM のデータは変化しません。

図 3.1.1 に TMP91FW40 リセットタイミングチャートを示します。

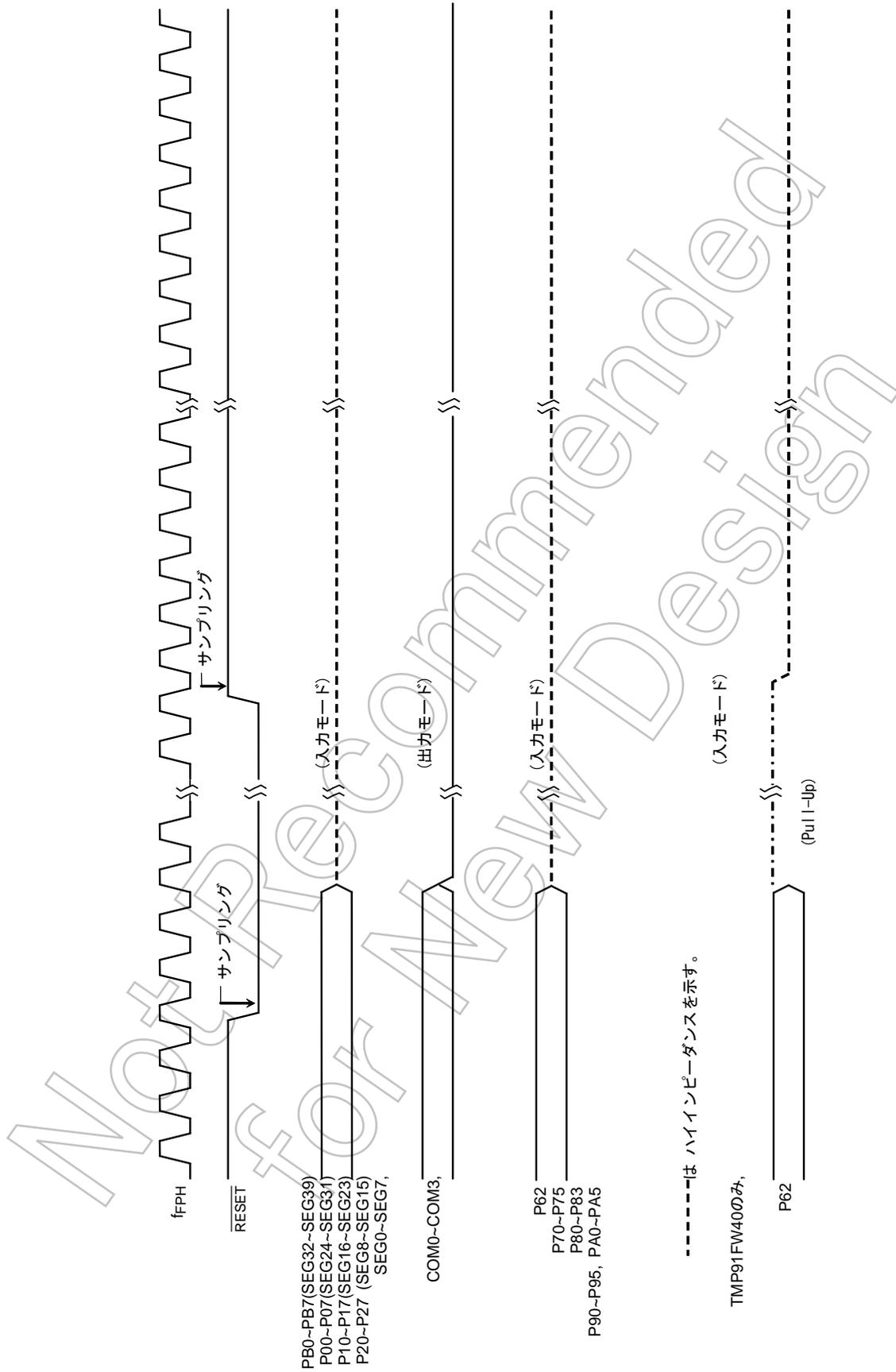


図 3.1.1 TMP91FW40 リセットタイミングチャート

### 3.1.2 動作モード概要

動作モードには、シングルチップモード、シングルブートモードがあります。各モードは、リセット解除時の端子状態により、設定されます。

- シングルチップモード: 通常動作を行うモードです。リセット解除後、内蔵フラッシュメモリプログラムの実行を開始します。
- シングルブートモード: 内蔵フラッシュメモリの書き換えをシリアル転送 (UART) で行うモードです。リセット解除後、内蔵ブート ROM が起動し、オンボード書き換えプログラムが実行されます。

表 3.1.1 動作モード設定表

動作モード	モード設定入力端子			
	RESET	BOOT (P62)	AM0	AM1
シングルチップ		H	H	H
シングルブート		L		

### 3.2 メモリマップ

TMP91FW40 のメモリマップを、図 3.2.1 に示します。

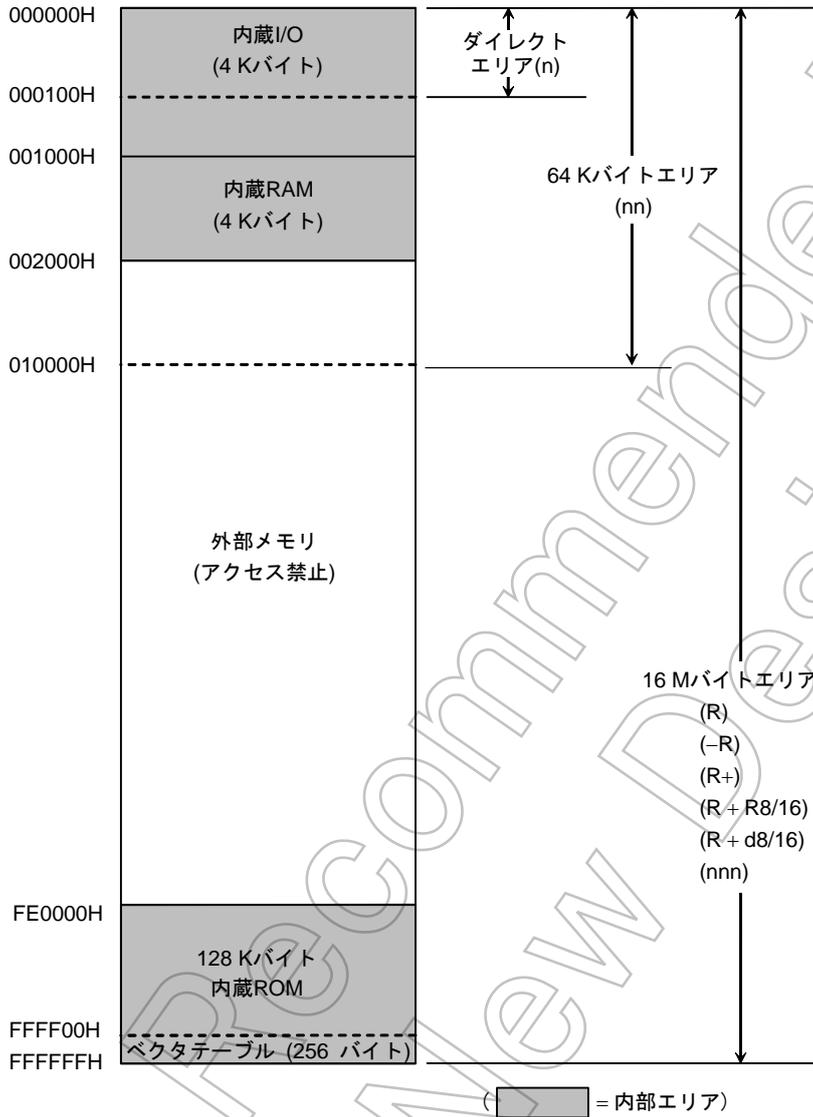


図 3.2.1 メモリマップ

### 3.3 フラッシュメモリ

TMP91FW40は3V単一電源による電氣的消去および書き込み可能なフラッシュメモリを内蔵しています。

フラッシュメモリの書き込みおよび消去は、JEDEC標準コマンドで行います。コマンド入力後、書き込みおよび消去が内部で自動的に行われます。また、消去動作は一括消去、セクタ単位での消去ができます。

内蔵フラッシュメモリの構成およびその動作を説明します。

#### 3.3.1 特長

- 書き込み/消去時の電源電圧  
Vcc = 2.7~3.6 V (-10°C~40°C)
- 構成  
64 K × 16 ビット (128 K バイト)
- 機能  
1ワード書き込み  
チップイレース  
セクタイレース  
データポーリング/トグルビット
- セクタサイズ  
4 K バイト × 32
- モードコントロール  
JEDEC標準コマンド準拠
- プログラミング方法  
オンボード書き込み  
パラレルライター書き込み
- セキュリティ  
ライトプロテクト  
リードプロテクト

#### 3.3.2 ブロック図

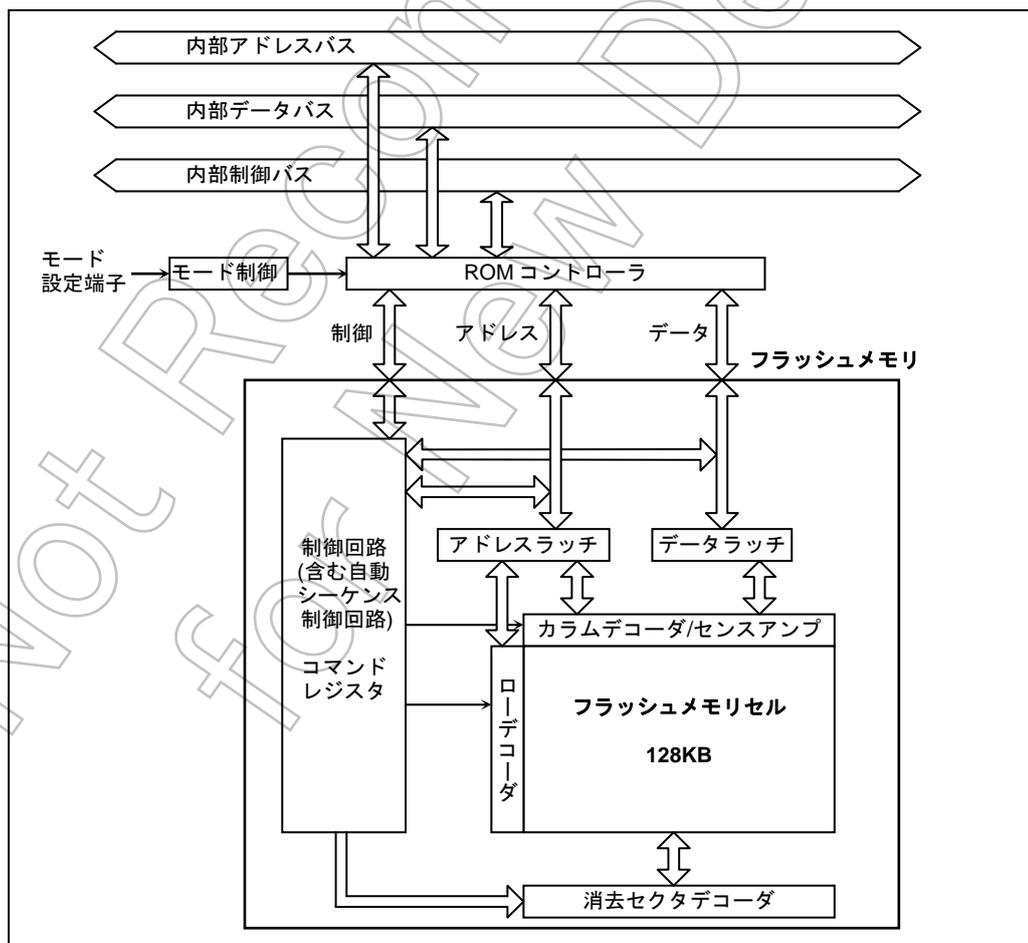


図 3.3.1 フラッシュ部ブロック図

### 3.3.3 動作モード

#### 3.3.3.1 概要

内蔵フラッシュメモリの書き込み/消去などの制御は、以下の3通りの動作状態(モード)が存在します。

表 3.3.1 動作モード説明

動作モード名	動作の内容
シングルチップモード	リセット解除後、内蔵のフラッシュメモリから起動します。 本動作モードの中で、ユーザのアプリケーションプログラムを実行するモードと、ユーザのセット上でフラッシュメモリの書き替えを実行するモードとに分けて定義します。前者を「ノーマルモード」、後者を「ユーザブートモード」と呼びます。 この両者の切り替えはユーザが独自に設定できます。 例えばポート 00 が “1” のときノーマルモード、“0” のときにユーザブートモードというように自由に設計することが可能です。 ユーザはアプリケーションプログラムの一部に切り替えを判定するためのルーチンを準備してください。
ノーマルモード	ユーザのアプリケーションプログラムより起動します。
ユーザブートモード	ユーザ指定の方法により内蔵フラッシュの書き替えを実行します。
シングルブートモード	リセット解除後、内蔵するブート ROM (Mask ROM) から起動します。ブート ROM には、シリアルポートを経由してユーザのセット上で書き込み/消去を行う為のプログラムを、デバイスの RAM 上に転送することができるアルゴリズムがプログラムされています。書き込みプログラムを RAM 上に搭載し、外部ホストから書き込みデータを受信しつつ、フラッシュへの書き込みコマンドを発行することで、フラッシュの書き込み/消去が実行できます。
ライターモード	汎用のプログラムライターで内蔵フラッシュメモリを書き込み/消去をするためのモードです。 プログラムライターのサポート状況については、当社営業窓口までお問い合わせください。

表 3.3.1のうち、ユーザブートモード、シングルブートモード、ライターモードの3つが内蔵フラッシュメモリの書き替えが可能な動作モードです。ユーザのセット上で書き込み/消去が可能なモードは、オンボードプログラミングモードと定義します。

オンボードプログラミングモードは、「シリアル I/O を利用した当社独自の書き込み/消去方式をサポートするシングルブートモード」、「シングルチップモード内においてユーザが独自に書き込み/消去方式を構築できるユーザブートモード」があります。

また、本デバイスはライターモード中に、ROM データの読み出しを禁止する「リードプロテクト機能」を持っています。プログラミング完了時にリードプロテクト機能をオンにしておくことで、第三者への ROM データ流出を阻止することができます。

シングルチップ、ブートおよびライタの各動作モードは、リセット状態で入力端子 AM0、AM1、 $\overline{\text{BOOT}}$  (P62)のレベルを外部で設定することにより決定されます。

CPU はライタモードを除き、状態設定後リセットを解除することにより各動作モードで動作を開始します。ライタモードは $\overline{\text{RESET}} = "0"$ のまま使用します。それぞれモード設定後は動作中にレベルの変更がないようにしてください。以下に各動作モードの設定方法とモード遷移図を示します。

表 3.3.2 動作モード設定表

	動作モード	入力端子			
		$\overline{\text{RESET}}$	$\overline{\text{BOOT}}$ (P62)	AM1	AM0
(1)	シングルチップモード (ノーマルおよびユーザブート)		1	1	1
(2)	シングルブートモード	0	0	1	1
(3)	ライタモード	0	—	1	0

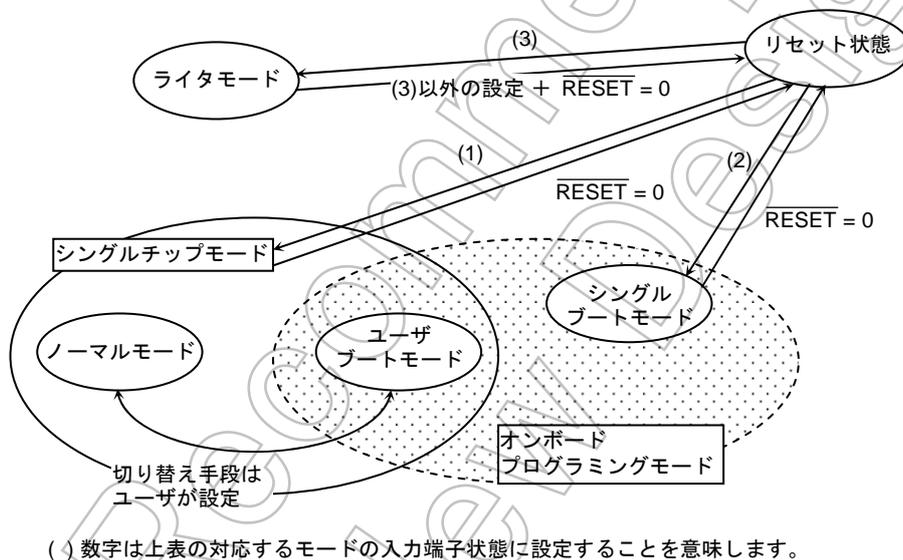


図 3.3.2 モード遷移図

### 3.3.3.2 リセット動作

本デバイスにリセットをかけるには、電源電圧が動作電圧範囲内で、かつ内部高周波発振器の発振が安定した状態で、少なくとも 10 システムクロック間  $\overline{\text{RESET}}$  入力を "0" にしてください。詳細は 3.1 項「CPU」の 3.1.1 「リセット動作」を参照してください。

### 3.3.3.3 モード別メモリマップ

本製品では動作モードごとにメモリマップが変わります。以下に、各動作モードのメモリマップとモード別セクタアドレス範囲表を示します。

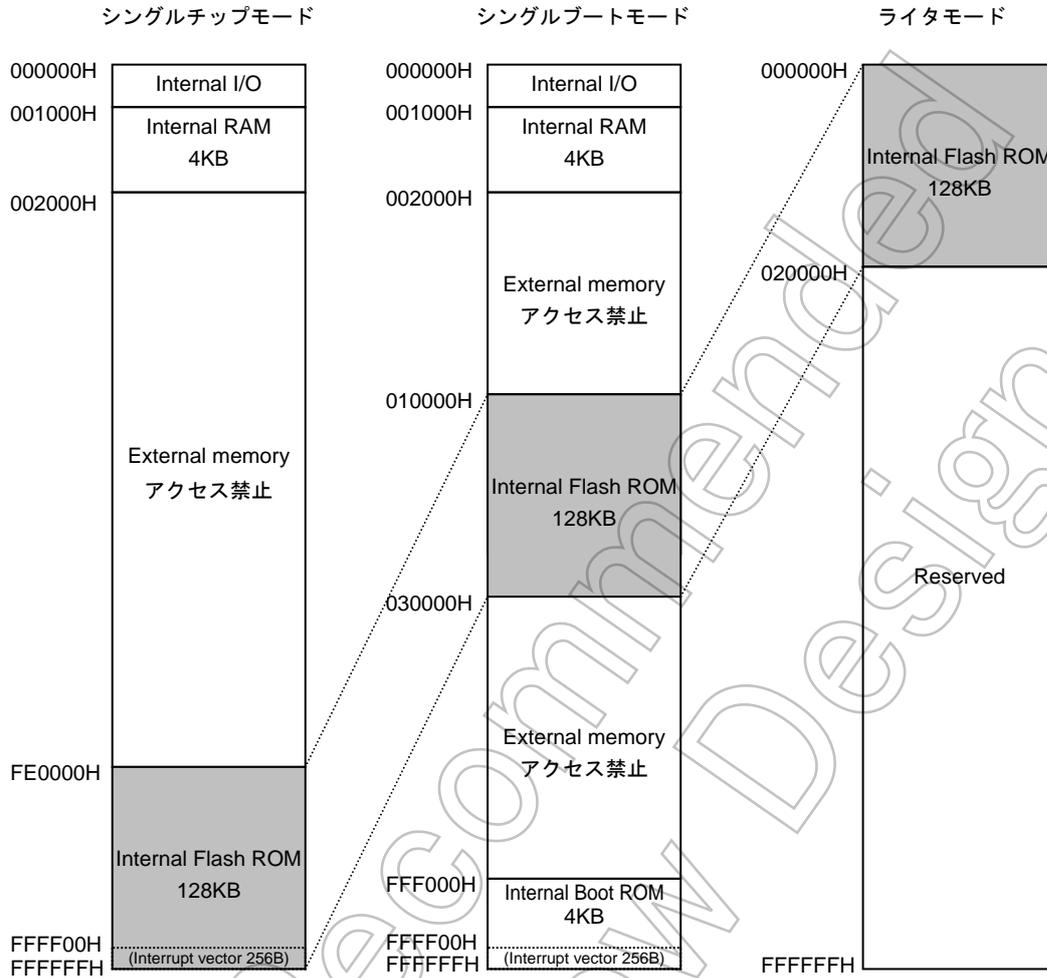


図 3.3.3 TMP91FW40 モード別メモリマップ

表 3.3.3 モード別セクタアドレス範囲表

	シングルチップモード	シングルブートモード
Sector-0	FE0000H から FE0FFFH まで	10000H から 10FFFH まで
Sector-1	FE1000H から FE1FFFH まで	11000H から 11FFFH まで
Sector-2	FE2000H から FE2FFFH まで	12000H から 12FFFH まで
Sector-3	FE3000H から FE3FFFH まで	13000H から 13FFFH まで
Sector-4	FE4000H から FE4FFFH まで	14000H から 14FFFH まで
Sector-5	FE5000H から FE5FFFH まで	15000H から 15FFFH まで
Sector-6	FE6000H から FE6FFFH まで	16000H から 16FFFH まで
Sector-7	FE7000H から FE7FFFH まで	17000H から 17FFFH まで
Sector-8	FE8000H から FE8FFFH まで	18000H から 18FFFH まで
Sector-9	FE9000H から FE9FFFH まで	19000H から 19FFFH まで
Sector-10	FEA000H から FEAFFFH まで	1A000H から 1AFFFH まで
Sector-11	FEB000H から FEBFFFH まで	1B000H から 1BFFFH まで
Sector-12	FEC000H から FECFFFH まで	1C000H から 1CFFFH まで
Sector-13	FED000H から FEDFFFH まで	1D000H から 1DFFFH まで
Sector-14	FEE000H から FEEFFFH まで	1E000H から 1EFFFH まで
Sector-15	FEF000H から FEFFFFH まで	1F000H から 1FFFFH まで
Sector-16	FF0000H から FF0FFFH まで	20000H から 20FFFH まで
Sector-17	FF1000H から FF1FFFH まで	21000H から 21FFFH まで
Sector-18	FF2000H から FF2FFFH まで	22000H から 22FFFH まで
Sector-19	FF3000H から FF3FFFH まで	23000H から 23FFFH まで
Sector-20	FF4000H から FF4FFFH まで	24000H から 24FFFH まで
Sector-21	FF5000H から FF5FFFH まで	25000H から 25FFFH まで
Sector-22	FF6000H から FF6FFFH まで	26000H から 26FFFH まで
Sector-23	FF7000H から FF7FFFH まで	27000H から 27FFFH まで
Sector-24	FF8000H から FF8FFFH まで	28000H から 28FFFH まで
Sector-25	FF9000H から FF9FFFH まで	29000H から 29FFFH まで
Sector-26	FFA000H から FFAFFFH まで	2A000H から 2AFFFH まで
Sector-27	FFB000H から FFBFFFH まで	2B000H から 2BFFFH まで
Sector-28	FFC000H から FFCFFFH まで	2C000H から 2CFFFH まで
Sector-29	FFD000H から FFDFFFH まで	2D000H から 2DFFFH まで
Sector-30	FFE000H から FFEFFFH まで	2E000H から 2EFFFH まで
Sector-31	FFF000H から FFFFFFH まで	2F000H から 2FFFFH まで

### 3.3.4 シングルブートモード

内蔵ブートROM (マスク ROM) を起動して、外部から書き込みルーチン (ユーザ作成のブートプログラム) をRAMへ転送し、そのルーチンプログラムを利用してフラッシュメモリを書き込み/消去する方法です。このモードでは、内蔵ブートROMが割り込みベクタテーブルを含む領域にマッピングされ、内蔵ブートROMプログラムが起動されます。また、フラッシュメモリはブートROM領域と別のアドレス空間にマッピングされます (図 3.3.3参照)。

本デバイスの SIO (SIO1) とコントローラを接続し、コントローラ側からデバイスの内蔵RAMに書き込みプログラムを転送し、RAM上の書き込みルーチンを実行してフラッシュメモリの書き込みを行います。

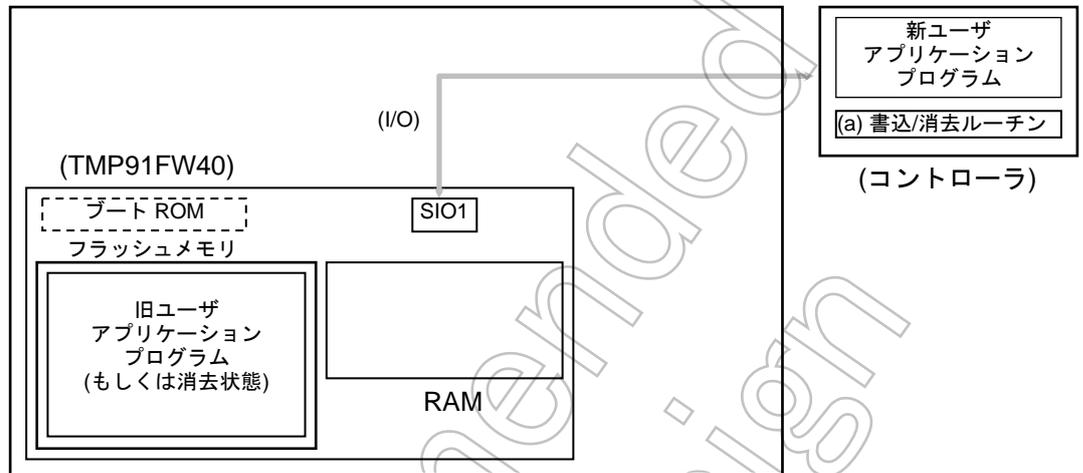
書き込みルーチンは、コントローラ側からコマンドおよび書き込みデータを送出することにより実行します。コントローラ側との通信の詳細は後述のプロトコルに従ってください。RAMへのプログラム転送は、ユーザのROMデータに対するセキュリティ確保のため、実行に先立ちユーザパスワードの照合を行います。パスワードが一致しない場合は、RAM転送そのものが実行されません。シングルブートモードでは、割り込み動作を禁止し、割り込み要求は割り込み要求フラグにて確認してください。

**注) シングルブートモード時、ブートROMのプログラムはNORMALモードで動作します。  
書き換えプログラムの中で他のモードへ遷移させないでください。**

## 3.3.4.1 内蔵ブートROMの書き込み/消去アルゴリズムを利用する場合

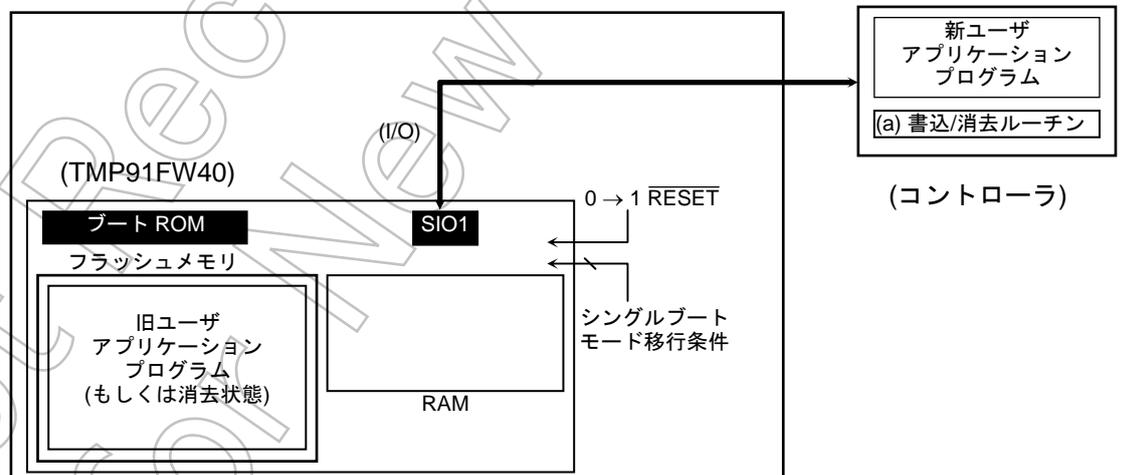
## (Step-1)環境準備

書き込み/消去ルーチン、書き替えデータなどの転送は SIO (SIO1)を経由して行いますので、ボード上で本デバイスの SIO (SIO1)と外部コントローラとをつなげます。書き込み/消去を行うための (a) 書き込み/消去ルーチンをコントローラ上に用意します。



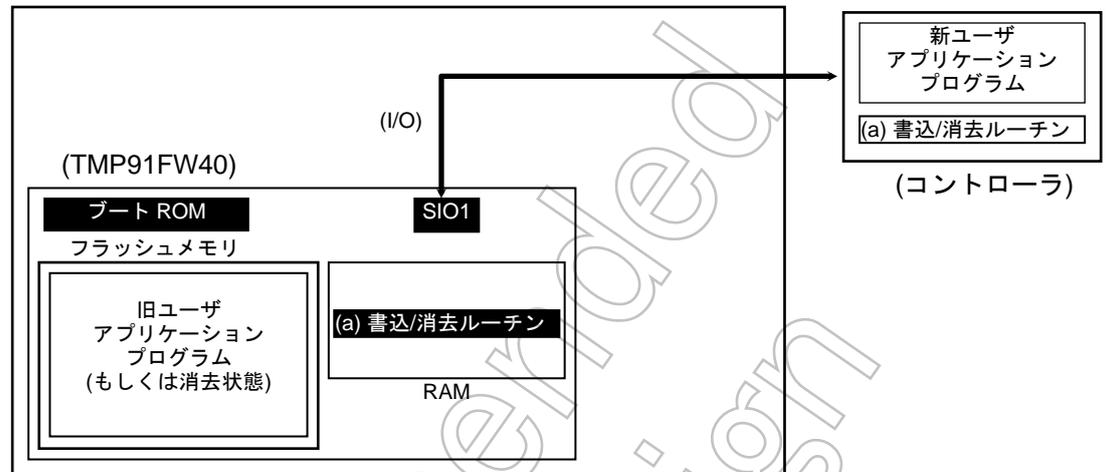
## (Step-2) ブートの起動(内蔵ブートROM起動)

ブートモードの端子条件設定でリセットを解除し、内蔵ブートROMを起動します。シングルブート動作の通信手順に従い、SIOを経由して転送元(コントローラ)より (a) 書き込み/消去ルーチンの転送を行います。ここでは、ユーザアプリケーションプログラム上に記録されているパスワードとの照合を行います。(フラッシュメモリが消去されている状態でも、消去データ("0xFF")12バイト長)をパスワードとして照合を行います。)



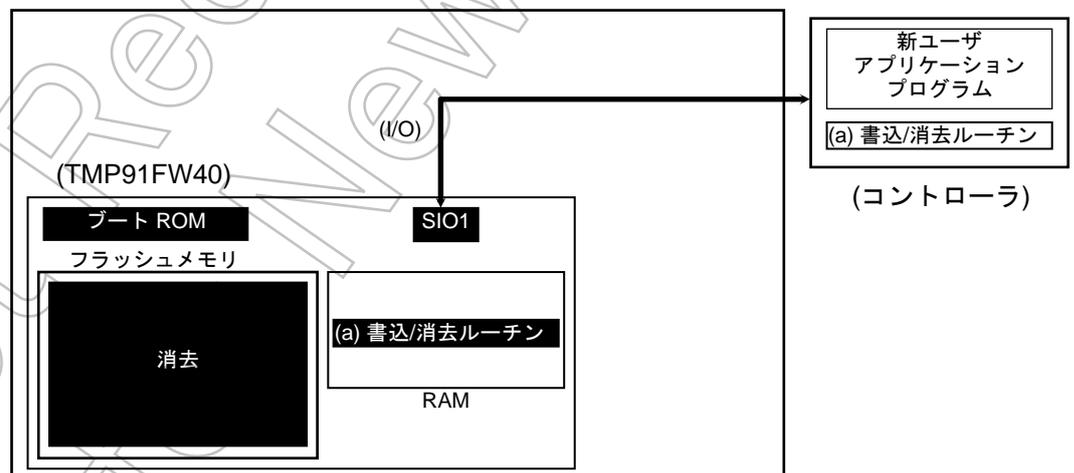
**(Step-3) RAM への書き替えルーチンのコピー**

パスワードの照合が終了すると、ブート ROM はシリアル通信を使用し、コントローラから(a)書き込み/消去ルーチンを内蔵 RAM へコピーします。ただし、RAM 上のアドレス 001000H~001DFFH の範囲に格納してください。

**(Step-4) RAM からの書き替えルーチンの実行**

RAM 上の (a) 書き込み/消去ルーチンへ制御を移し、消去が必要な場合は、旧ユーザアプリケーションプログラムエリアの消去を行ってください(セクタ単位もしくはチップ消去)。

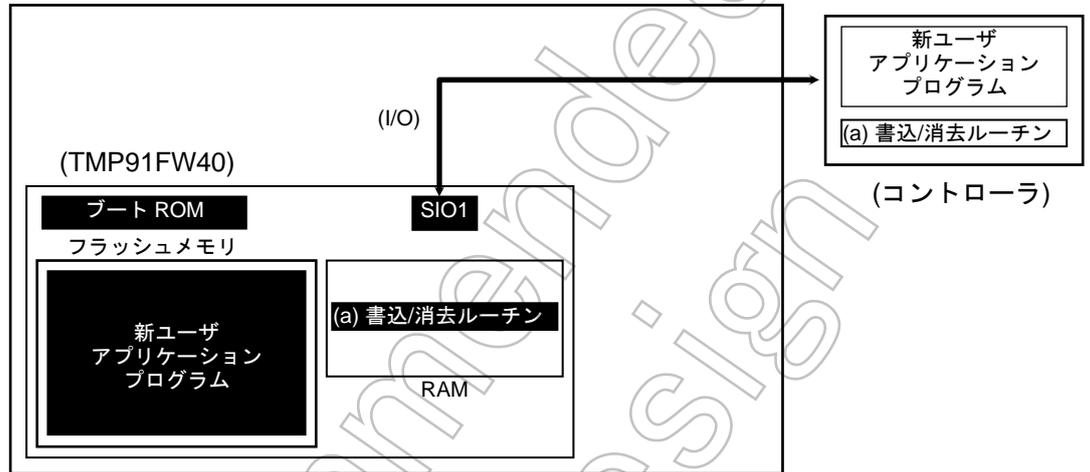
注)内蔵ブートは、消去コマンドを持っているため、書き込み/消去ルーチンを用いずに、コントローラからチップ消去が可能です。セクタ消去を行う場合は、書き込み/消去ルーチン上に必要なプログラムを組み込んでください。



(Step-5) 新ユーザアプリケーションのコピー

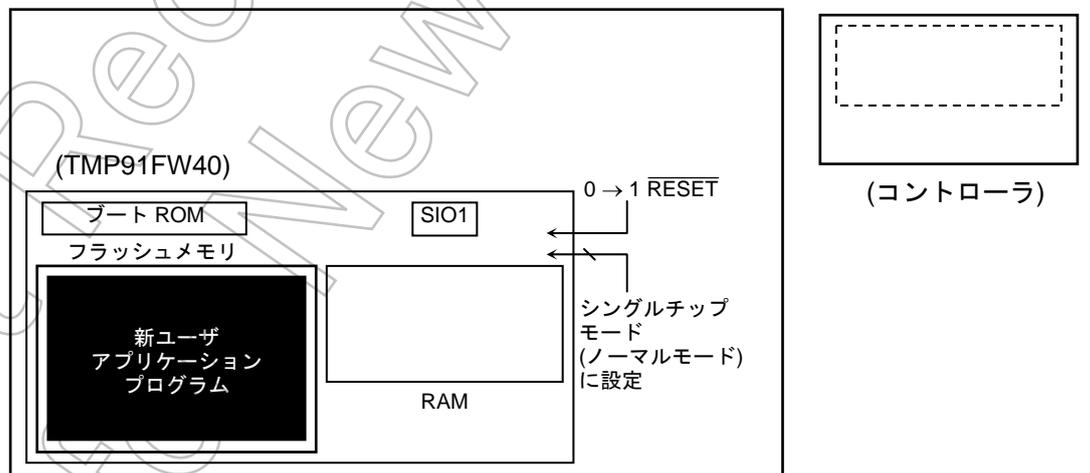
さらに、RAM 上の (a) 書き込み/消去ルーチンを実行して、転送元 (コントローラ) より新ユーザアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。

下の例では、書き替えデータ転送時には、書き替え/消去ルーチンを転送したときと同様の通信設定を使用していますが、書き替え/消去ルーチンを転送後は、転送したルーチンを用いて、設定 (データバスおよび転送元) を変更できます。必要に応じて、ボードのハードおよび書き込み/消去ルーチンを組み立ててください。



(Step-6) 新ユーザアプリケーションプログラムの起動

書き込みが完了したら、一度ボードの電源を落とし、転送元 (コントローラ) と接続していたケーブルをはずします。その後、再度電源を入れ直し、シングルチップモード (ノーマルモード) に設定し、新しいユーザアプリケーションプログラムを実行します。



### 3.3.4.2 シングルブートモードでの接続例

シングルブートモードでは、シリアル転送によるフラッシュメモリの書き替えを行います。したがって、オンボードプログラミングは本デバイスのSIO (チャンネル 1)とコントローラ (書き込みツール) を接続し、コントローラ側からコマンドを送出することにより実行します。図 3.3.4に書き込みコントローラとターゲットボードの接続例を、図 3.3.5にRS232Cボードとターゲットボードの接続例を示します。

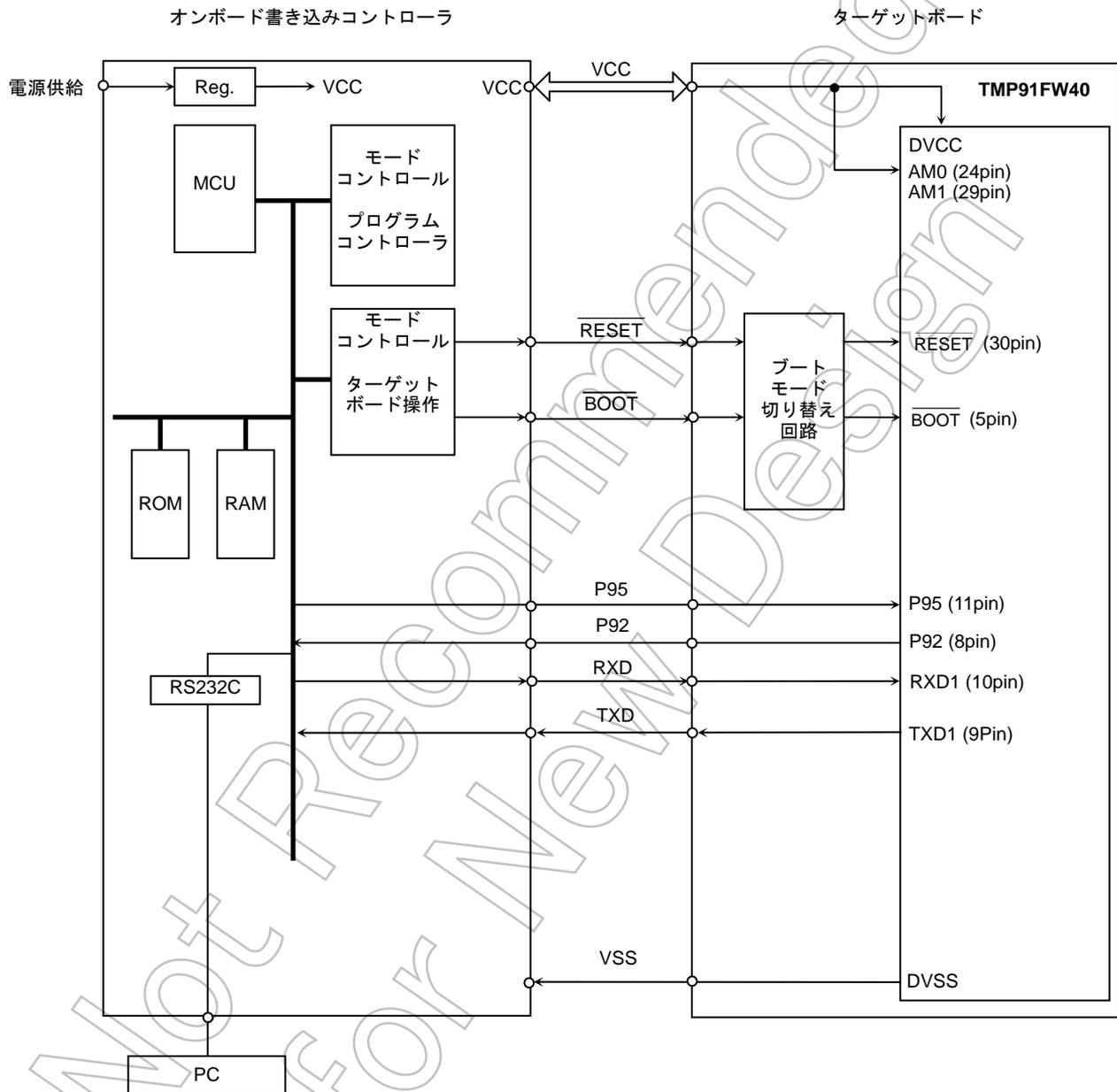


図 3.3.4 シングルブートモードでの外部コントローラとの接続例

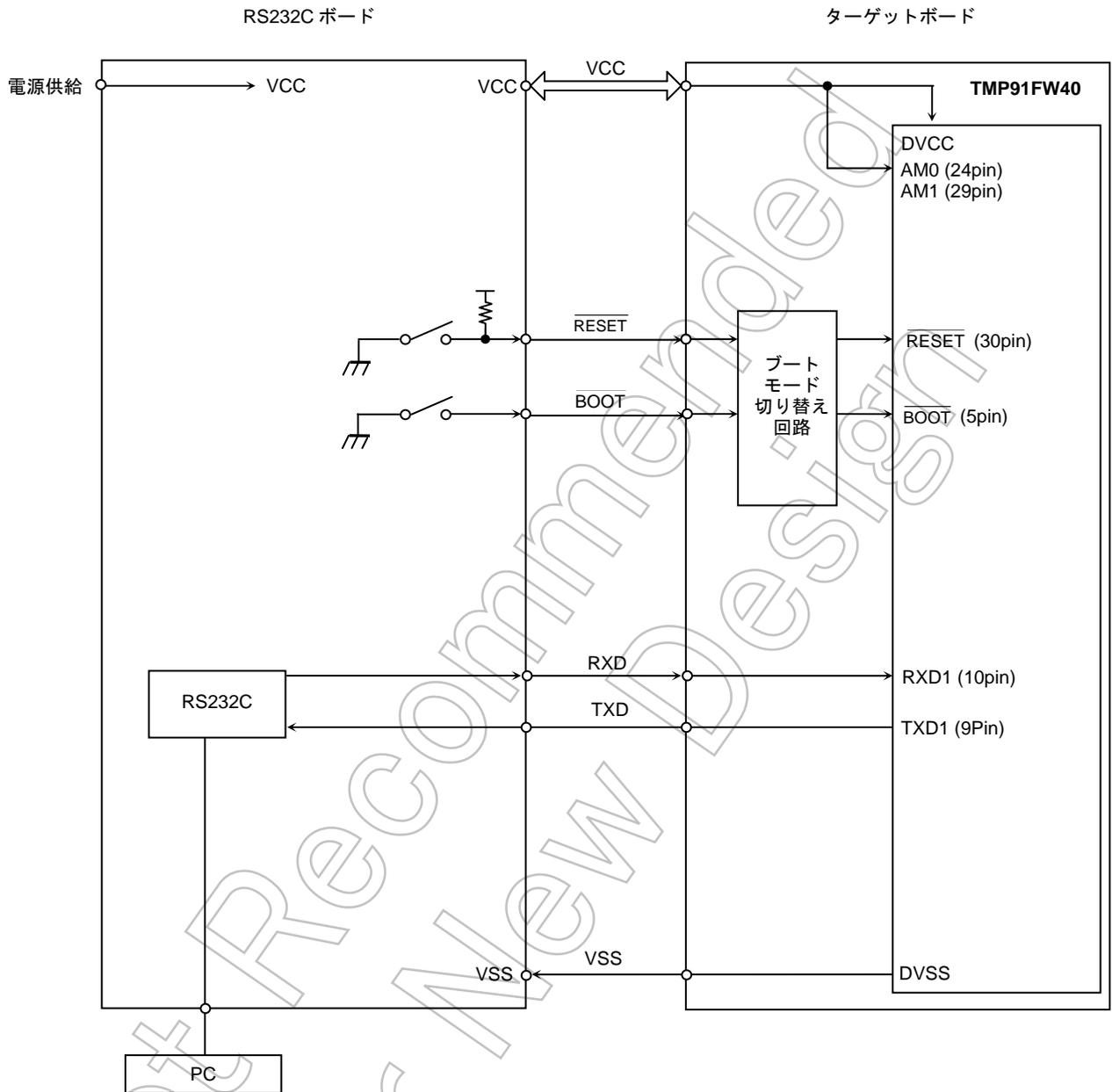


図 3.3.5 シングルブートモードでの RS232C ボードとの接続例

### 3.3.4.3 モード設定

オンボードプログラミングを実行するためには、本デバイスをシングルブートモードで立ち上げます。シングルブートモードで立ち上げるための設定を以下に示します。

- ・ AM0,AM1 = 1
- ・  $\overline{\text{BOOT}}$  = 0
- ・  $\overline{\text{RESET}}$  = 0 → 1

$\overline{\text{RESET}}$  入力端子を“0”の状態にして、AM0、AM1、 $\overline{\text{BOOT}}$ の各端子をあらかじめ上記条件に設定します。その後、RESET 解除を行うとシングルブートモードで起動します。

### 3.3.4.4 メモリマップ

図 3.3.6にノーマルモードとシングルブートモードのメモリマップの比較を示します。図 3.3.6のように、シングルブートモードでは、内蔵フラッシュメモリは10000H~2FFFFH番地(物理アドレス)にマッピングされます。また、FFF000H番地からFFFFFFFH番地にはブートROM (マスクROM) がマッピングされます。

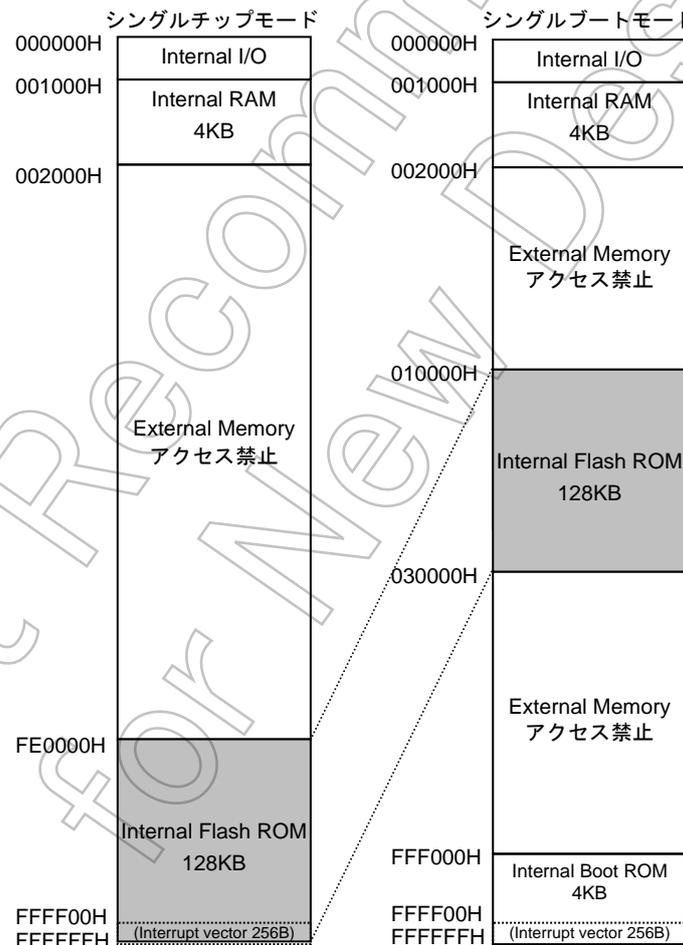


図 3.3.6 メモリマップの比較

### 3.3.4.5 インタフェース仕様

シングルブートモードでの SIO の通信フォーマットを以下に示します。シリアル動作のモードは、UART (非同期通信)に対応しています。

オンボードプログラミングを実行するためには、書き込みコントローラ側の通信フォーマットも同様に設定する必要があります。

#### UART(非同期)通信

- ・通信チャンネル : SIOチャンネル1 (使用端子は表 3.3.4 参照)
- ・シリアル転送モード : UART (非同期通信) モード
- ・データ長 : 8 ビット
- ・パリティビット : なし
- ・STOP ビット : 1 ビット
- ・ボーレート : 表 3.3.5、表 3.3.6 参照

表 3.3.4 端子の接続

端 子		UART
電源系端子	DVCC	○
	DVSS	○
モード設定端子	AM1,AM0, BOOT	○
リセット端子	RESET	○
通信端子	TXD1	○
	RXD1	○

注) 未使用端子は、リセット解除後の初期状態になっています。

表 3.3.5 ボーレート表

SIO	転送レート(bps)				
UART	115200	57600	38400	19200	9600

表 3.3.6 シングルブートモード時の動作周波数とボーレート対応表

基準ボーレート (bps)		9600		19200		38400		57600		115200	
基準周波数 (MHz)	対応範囲 (MHz)	ボーレート (bps)	備考								
8.0	7.84~8.16	○	注1	○	注1	○	注1	○	注2	×	—
8.0~9.6304	7.84~10.02	○	注1	○	注1	○	注1	×	—	×	—
8.0~19.6608	7.84~20.05	×	—	○	注1	○	注1	×	—	×	—
8.0~27.0	7.84~27.54	×	—	×	—	○	注1	×	—	×	—
11.0592~14.0	10.84~14.28	×	—	○	注1	○	注1	○	注1	×	—
11.0592~27.0	10.84~27.54	×	—	×	—	○	注1	○	注1	×	—
14.7456	14.46~15.04	×	—	○	注1	○	注1	○	注1	○	注1
16.0~18.4320	15.68~18.80	×	—	○	注1	○	注1	○	注1	○	注2
20.0	19.60~20.40	×	—	×	—	○	注1	○	注1	○	注2
22.1184~27.0	21.68~27.54	×	—	×	—	○	注1	○	注1	○	注1

基準周波数: シングルブートで対応可能な高速発振回路の周波数。

シングルブートモードを使ってフラッシュメモリの書き替えを行う場合は、高速クロックとして基準周波数のいずれかを選択してください。

対応範囲: 各基準周波数として検出されるクロック周波数の目安です。この範囲に含まれないクロック周波数では、シングルブート動作が行えない場合があります。

ボーレート: ○=通信可能、×=通信不可能

注1: 基準周波数 (マイコンクロック周波数) の自動検出を行うために、フラッシュメモリ書き替えコントローラの送信ボーレートの送信周波数誤差との総合誤差を、±3%以内にしてください。

注2: 基準周波数 (マイコンクロック周波数) の自動検出を行うために、フラッシュメモリ書き替えコントローラの送信ボーレートの送信周波数誤差との総合誤差を、±2%以内にしてください。

### 3.3.4.6 データ転送フォーマット

動作コマンド、および各動作モード時のデータ転送フォーマットをそれぞれ表 3.3.7~表 3.3.13に示します。

表 3.3.7 動作コマンドデータ

動作コマンドデータ	動作モード
10H	RAM 転送
20H	フラッシュメモリ SUM
30H	製品情報読み出し
40H	フラッシュメモリチップ消去
60H	フラッシュメモリプロテクト設定

Not Recommended  
for New Design

表 3.3.8 シングルブートプログラムの転送フォーマット [RAM 転送の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ	
BOOT ROM	1 バイト目	ボーレート設定 UART	86H	—	
	2 バイト目	—	所望の ボーレート (注 1)	ボーレート設定に対する ACK 応答 正常 (設定可能) の場合 ・UART 86H (ボーレートの設定が不可能と判断した 場合は動作停止)	
	3 バイト目	動作コマンドデータ (10H)		—	
	4 バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 x1H プロテクト設定済みの場合 (注 4) x6H 通信異常の場合 x8H	
	5 バイト目 ~ 16 バイト目	PASS WORD データ (12 バイト) (02FEF4H~02FEFFH)		—	
	17 バイト目	5 ~ 16 バイト目の CHECK SUM 値		—	
	18 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 11H 通信異常の場合 18H	
	19 バイト目	RAM 格納開始アドレス 31 ~ 24 (注 3)		—	
	20 バイト目	RAM 格納開始アドレス 23 ~ 16 (注 3)		—	
	21 バイト目	RAM 格納開始アドレス 15 ~ 8 (注 3)		—	
	22 バイト目	RAM 格納開始アドレス 7 ~ 0 (注 3)		—	
	23 バイト目	RAM 格納バイト数 15 ~ 8 (注 3)		—	
	24 バイト目	RAM 格納バイト数 7 ~ 0 (注 3)		—	
	25 バイト目	19~24 バイト目の CHECK SUM 値 (注 3)		—	
	26 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 11H 通信異常の場合 18H	
	27 バイト目 ~ m バイト目	RAM 格納データ		—	
	m+1 バイト目	27 ~ m バイト目の CHECK SUM 値		—	
	m+2 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 11H 通信異常の場合 18H	
	RAM	m+3 バイト目		—	JUMP RAM 格納開始アドレス

注 1) 所望のボーレートは表 3.3.6 を参照してください。

注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。

注 3) 19 バイト目~25 バイト目のデータは、RAM 上のアドレス 001000H~001DFFH (3.5KB) の領域内に納まるようにプログラムしてください。

注 4) リードプロテクトまたはライトプロテクトが設定されている場合は、受信したコマンドの動作を中止し、次の動作コマンド (3 バイト目) 待ちになります。

表 3.3.9 シングルブートプログラムの転送フォーマット [フラッシュメモリ SUM の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
BOOT ROM	1 バイト目	ボーレート設定 UART	86H	—
	2 バイト目	—	所望の ボーレート (注 1)	ボーレート設定に対する ACK 応答 正常 (設定可能) の場合 ・UART 86H (ボーレートの設定が不可能と判断した 場合は動作停止)
	3 バイト目	動作コマンドデータ (20H)		—
	4 バイト目	—	動作コマンドに対する ACK 応答 (注 2) 正常の場合 20H 異常の場合 x1H 通信異常の場合 x8H	—
	5 バイト目	—		SUM (上位)
	6 バイト目	—		SUM (下位)
	7 バイト目	—		5 ~ 6 バイト目の CHECK-SUM 値
	8 バイト目	(次の動作コマンドデータ待ち)		—

注 1) 所望のボーレートは表 3.3.6 を参照してください。

注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。

表 3.3.10 シングルブートプログラムの転送フォーマット [製品情報読み出しの場合] (1/2)

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
BOOT ROM	1バイト目	ボーレート設定 UART	86H	—
	2バイト目	—	所望の ボーレート (注 1)	ボーレート設定に対する ACK 応答 正常 (設定可能) の場合 UART 86H (ボーレートの設定が不可能と判断した 場合は、動作停止)
	3バイト目	動作コマンドデータ (30H)		—
	4バイト目	—	動作コマンドに対する ACK 応答 (注 2)	正常の場合 30H 異常の場合 x1H 通信異常の場合 x8H
	5バイト目	—		フラッシュメモリデータ (02FEF0H 番地)
	6バイト目	—	フラッシュメモリデータ (02FEF1H 番地)	
	7バイト目	—	フラッシュメモリデータ (02FEF2H 番地)	
	8バイト目	—	フラッシュメモリデータ (02FEF3H 番地)	
	9バイト目	—	製品名 (アスキーコード、12 バイト)	
	~	—	9 バイト目から 'TMP91FW40_ _ _'	
	20バイト目	—	Password 比較開始アドレス (4 バイト)	
	21バイト目	—	21 バイト目から F4H, FEH, 02H, 00H	
	~	—		
	24バイト目	—	RAM 開始アドレス (4 バイト)	
	25バイト目	—	25 バイト目から 00H, 10H, 00H, 00H	
	~	—		
	28バイト目	—	RAM(ユーザ領域)終了アドレス (4 バイト)	
	29バイト目	—	29 バイト目から FFH, 1DH, 00H, 00H	
	~	—		
	32バイト目	—	RAM 終了アドレス (4 バイト)	
	33バイト目	—	33 バイト目から FFH, 1FH, 00H, 00H	
	~	—		
	36バイト目	—	ダミーデータ (4 バイト)	
	37バイト目	—	37 バイト目から 00H, 00H, 00H, 00H	
	~	—		
	40バイト目	—	ダミーデータ (4 バイト)	
	41バイト目	—	41 バイト目から 00H, 00H, 00H, 00H	
	~	—		
	44バイト目	—	FUSE 情報 (2 バイト) 45 バイト目から リード/ライトプロテクト設定	
	45バイト目	—	1) あり / あり : 00H, 00H	
~	—	2) なし / あり : 01H, 00H		
46バイト目	—	3) あり / なし : 02H, 00H		
~	—	4) なし / なし : 03H, 00H		
47バイト目	—	フラッシュメモリ開始アドレス (4 バイト)		
~	—	47 バイト目から 00H, 00H, 01H, 00H		
50バイト目	—	フラッシュメモリ終了アドレス (4 バイト)		
51バイト目	—	51 バイト目から FFH, FFH, 02H, 00H		
~	—			
54バイト目	—	フラッシュメモリセクタ分割数情報 (2 バイト)		
55バイト目	—	55 バイト目から 20H, 00H		
~	—			
56バイト目	—	フラッシュメモリ同一セクタサイズの 開始アドレス (4 バイト)		
57バイト目	—	57 バイト目から 00H, 00H, 01H, 00H		
~	—			
60バイト目	—			

表 3.3.11 シングルブートプログラムの転送フォーマット [製品情報読み出しの場合] (2/2)

	転送バイト数	コントローラ→本デバイスへの 転送データ	ポーレート	本デバイス→コントローラへの 転送データ
BOOT ROM	61 バイト目 ~ 64 バイト目	—		フラッシュメモリ同一セクタサイズの サイズ (ハーフワード) (4 バイト) 61 バイト目から 00H, 08H, 00H, 00H
	65 バイト目	—		フラッシュメモリ同一セクタサイズの 個数 (1 バイト) 20H
	66 バイト目	—		5 ~ 65 バイト目の CHECK SUM 値
	67 バイト目	(次の動作コマンドデータ待ち)		—

- 注 1) 所望のポーレートは表 3.3.6 を参照してください。  
 注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。

Not Recommended  
for New Design

表 3.3.12 シングルブートプログラムの転送フォーマット [フラッシュメモリチップ消去の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
BOOT ROM	1 バイト目	ボーレート設定 UART	86H	—
	2 バイト目	—	所望の ボーレート (注 1)	ボーレート設定に対する ACK 応答 正常 (設定可能) の場合 ・UART 86H (ボーレートの設定が不可能と判断した 場合は動作停止)
	3 バイト目	動作コマンドデータ (40H)		—
	4 バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 40H 異常の場合 x1H 通信異常の場合 x8H
	5 バイト目	消去イネーブルコマンドデータ (54H)		—
	6 バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 54H 異常の場合 x1H 通信異常の場合 x8H
	7 バイト目	—		消去コマンドに対する ACK 応答 正常の場合 4FH 異常の場合 4CH
	8 バイト目	—		ACK 応答 正常の場合 5DH 異常の場合 60H
	9 バイト目	(次の動作コマンドデータ待ち)		—

- 注 1) 所望のボーレートは表 3.3.6 を参照してください。  
 注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。

表 3.3.13 シングルブートプログラムの転送フォーマット [フラッシュメモリプロテクト設定の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
BOOT ROM	1 バイト目	ボーレート設定 UART	86H	—
	2 バイト目	—	所望の ボーレート (注 1)	ボーレート設定に対する ACK 応答 正常 (設定可能) の場合 ・UART 86H (ボーレートの設定が不可能と判断した 場合は動作停止)
	3 バイト目	動作コマンドデータ (60H)		—
	4 バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 60H 異常の場合 x1H 通信異常の場合 x8H
	5 バイト目 ~ 16 バイト目	PASS WORD データ (12 バイト) (02FEF4H~02FEFFH)		—
	17 バイト目	5 ~ 16 バイト目の CHECK SUM 値		—
	18 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 60H 異常の場合 61H 通信異常の場合 68H
	19 バイト目	—		プロテクト設定コマンドに対する ACK 応答 正常の場合 6FH 異常の場合 6CH
	20 バイト目	—		ACK 応答 正常の場合 31H 異常の場合 34H
	21 バイト目	(次の動作コマンドデータ待ち)		—

注 1) 所望のボーレートは表 3.3.6 を参照してください。

注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。

### 3.3.4.7 ブートプログラム

シングルブートモード立ち上げ時にはブートプログラムが起動します。

ここではシングルブートモード起動時の内蔵ブートプログラムがコントローラと通信を行うのに必要なタイミングを述べます。ユーザがシングルブートを使用するコントローラを作成する場合や、ユーザがユーザブートの環境を構築する為の情報として活用ください。

#### 1. RAM 転送コマンド

RAM 転送は、コントローラから送られてくるデータを内蔵 RAM へ格納します。転送が正常に終了するとユーザプログラムの実行を開始します。ユーザプログラムのサイズは、最大 3.8K バイトまで転送可能です(スタックポインタエリア等の保護のため、ブートプログラム上で制限をかけています)。また、実行開始アドレスは、RAM 格納開始アドレスになります。

このRAM転送機能でユーザ作成の書き込み/消去プログラムを実行することにより、ユーザ独自のオンボードプログラミング制御を行うことができます。ユーザプログラムでオンボードプログラミングを実行するためには、3.3.6「フラッシュメモリコマンドシーケンス」を使う必要があります。RAM転送コマンドが終了した後は、内蔵RAMの全領域を使用することができます。

なお、デバイスにリードプロテクトまたはライトプロテクトが設定されている場合は、本コマンドは実行されません。また、パスワードエラーが発生した場合も、本コマンドは実行されません。

#### 2. フラッシュメモリ SUM コマンド

フラッシュメモリ 128K バイトの SUM を計算しその結果を返します。ブートプログラムではフラッシュメモリの全エリアのデータを読み出す動作コマンドはサポートしていません。その代わりに、このフラッシュメモリ SUM コマンドがあります。SUM を読み出すことで、アプリケーションプログラムのレビジョンを管理することができます。

#### 3. 製品情報読み出しコマンド

本デバイスの製品名やメモリ情報などを返します。製品情報読み出しコマンドでは、フラッシュメモリの一部エリア (02FEF0H~02FEF3H) のデータを返します。フラッシュメモリ SUM コマンド以外に、このデータを用いることでも、アプリケーションプログラムのレビジョンを管理することができます。

#### 4. フラッシュメモリチップ消去コマンド

すべてのセクタのフラッシュメモリを消去します。リード/ライトプロテクトがかかっている場合、メモリセルの全てのセクタを消去し、リード/ライトプロテクトを解除します。本コマンドは、パスワードを忘れた場合のブートプログラムの操作を回復する機能も兼用しているため、パスワード比較は行っていません。

#### 5. フラッシュメモリプロテクト設定コマンド

リードプロテクトとライトプロテクトを同時に設定します。但し、パスワードエラーが発生した場合は、本コマンドは実行されません。

リードプロテクトを設定するとライターモードの時にフラッシュメモリのリードができなくなります。また、ライトプロテクトを設定するとライターモードの時にフラッシュメモリのライトが出来なくなります。

Not Recommended  
for New Design

### 3.3.4.8 RAM転送コマンド (表 3.3.8参照)

#### 1. コントローラ → デバイス

1 バイト目のデータは、ボーレートを判定するデータになります。1 バイト目のデータは、受信を禁止した状態 ( $SC1MOD0<RXE> = 0$ ) にしています。

- UART で通信

コントローラからターゲットボードへは、UART の設定で、所望のボーレートでデータを 86H にして送信してください。シリアル動作モードの判定で UART と判定した場合、ボーレートの設定が可能かどうかを判定します。ボーレートが適切では無く、設定が不可能と判断した場合は動作を停止するため、通信が行えなくなります。

#### 2. デバイス → コントローラ

2 バイト目の送信データは、1 バイト目のシリアル動作モード設定データに対する ACK 応答データになります。1 バイト目のデータが、UART と判定されボーレートの設定が可能な場合 86H を送信します。

- ボーレート判定

ボーレートの設定が可能かどうかを判定します。設定が可能と判定した場合、内蔵ブートプログラムが、BR1CR、BR1ADD の値を書き替え、86H を送信し、設定が不可能と判定した場合、動作を停止するため何も送信しません。コントローラは、1 バイト目のデータの送信が終了した後、タイムアウト時間 (5 秒) を設けます。タイムアウト時間内に、データ (86H) を正常受信できなければ、通信不能と判断してください。受信を許可 ( $SC1MOD0<RXE> = 1$ ) するタイミングは、送信バッファにデータ (86H) を書き込む前に行っています。

#### 3. コントローラ → デバイス

3 バイト目の受信データは、動作コマンドデータになります。この場合は、RAM 転送コマンドデータ (10H) になります。

#### 4. デバイス → コントローラ

4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

次に、3 バイト目の受信データが、表 3.3.7 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。RAM 転送コマンドを選択した場合には、プロテクト (リードまたはライト) が設定されていないことをチェックし、10H をエコーバック送信して RAM 転送処理ルーチンに分岐します。プロテクトが設定されている場合は、ACK 応答データ (bit 2/1) x6H を送信して、再び動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

RAM 転送処理ルーチンに分岐後、パスワードエリアのデータをチェックします。パスワードエリアのデータのチェック方法は、3.3.4.14 「パスワードについて」を参照してください。

3 バイト目の受信データがいずれのコマンドにも該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、再び動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

## 5. コントローラ → デバイス

5 バイト目~16 バイト目の受信データは、パスワードデータ (12 バイト) になります。5 バイト目の受信データはフラッシュメモリの 02FEF4H のデータと照合し、6 バイト目の受信データはフラッシュメモリの 02FEF5H のデータと照合します。同様に 16 バイト目の受信データはフラッシュメモリの 02FEFFH のデータと照合します。

## 6. コントローラ → デバイス

17 バイト目の受信データは、CHECK SUMデータになります。5 バイト目から 16 バイト目の送信データを符号なしの 8 ビット加算 (オーバフローを無視) して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUMデータの計算方法は、3.3.4.16 「CHECK SUMの計算方法」を参照してください。

## 7. デバイス → コントローラ

18 バイト目の送信データは、5 バイト目~17 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。最初に、5 バイト目~17 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 18H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので、「1」になります。

次に、17 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、5 バイト目~17 バイト目までの受信データを符号なし 8 ビット加算 (オーバフローを無視) して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

最後に、パスワードの照合結果をチェックします。5 バイト目~16 バイト目のパスワードデータの照合がすべて一致しない場合、パスワードエラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

上記のチェックを終えて、すべて正常なら、正常 ACK 応答データ 10H を送信します。

## 8. コントローラ → デバイス

19 バイト目~22 バイト目までの受信データは、ブロック転送における格納先の RAM の開始アドレスを表します。19 バイト目がアドレスの 31 ビット~24 ビットに対応、20 バイト目がアドレスの 23 ビット~16 ビットに対応、21 バイト目が 15 ビット~8 ビット、22 バイト目が 7 ビット~0 ビットに対応します。

## 9. コントローラ → デバイス

23 バイト目、24 バイト目の受信データは、ブロック転送するバイト数を表します。23 バイト目が転送バイト数の 15 ビット~8 ビット目に対応し、24 バイト目が 7 ビット~0 ビット目に対応します。

## 10. コントローラ → デバイス

25 バイト目の受信データは、CHECK SUMデータになります。19 バイト目から 24 バイト目の送信データを符号なし 8 ビット加算 (オーバフローを無視) して得られた下位 8 ビット値の 2 の補数値をコントローラから送信してください。CHECK SUMデータ計算方法は、3.3.4.16 「CHECK SUMの計算方法」を参照してください。

注)19 バイト目~25 バイト目のデータは RAM 上のアドレス 001000H~001DFFH (3.5KB) の領域に納まるようにプログラムしてください。

## 11. デバイス → コントローラ

26 バイト目の送信データは、19 バイト目~25 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。

最初に、19 バイト目 ~ 25 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 18H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので“1”になります。

次に、25 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、19 バイト目 ~ 25 バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 10H を送信します。

## 12. コントローラ → デバイス

27 バイト目~m バイト目の受信データは、RAM へ格納するデータになります。RAM に格納するデータを、19 バイト目から 22 バイト目で指定されたアドレスから書き込み、23 バイト目から 24 バイト目に指定されたバイト数分だけ書き込みます。

## 13. コントローラ → デバイス

m+1 バイト目の受信データは、CHECK SUM データになります。27 バイト目 ~ m バイト目の送信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUM データの計算方法は、3.3.4.16 「CHECK SUM の計算方法」を参照してください。

## 14. デバイス → コントローラ

m+2 バイト目の送信データは、27 バイト目 ~ (m+1) バイト目のデータに対する ACK 応答データ (CHECK SUM に対する ACK 応答) になります。

最初に 27 バイト目 ~ m+1 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 18H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので“1”になります。

次に、m+1 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、27 バイト目 ~ m+1 バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 10H を送信します。

## 15. デバイス → コントローラ

m+2 バイト目の ACK 応答データが正常 ACK 応答データの場合、正常 ACK 応答データ 10H を送信後、19 バイト目 ~ 22 バイト目で指定された RAM のスタートアドレスに分岐します。

## 3.3.4.9 フラッシュメモリSUMコマンド (表 3.3.9参照)

1. 1バイト目～2バイト目までの送受信データはRAM転送コマンドの場合と同一になります。
2. コントローラ → デバイス  
3バイト目の受信データは動作コマンドデータになります。この場合は、フラッシュメモリSUMコマンドデータ(20H)になります。
3. デバイス → コントローラ  
4バイト目の送信データは、3バイト目の動作コマンドデータに対するACK応答データになります。  
最初に、3バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常のACK応答データ(bit 3) x8Hを送信して、次の動作コマンド(3バイト目)データ待ちになります。送信データの上位4ビットは不定値になります(直前の動作コマンドデータの上位4ビットになります)。  
次に、3バイト目の受信データが、表3.3.7の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信(正常ACK応答データ)します。この場合、20Hをエコーバック送信して、フラッシュメモリSUM処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーのACK応答データ(bit 0) x1Hを送信して、次の動作コマンド(3バイト目)データ待ちになります。送信データ上位4ビットは不定値になります(直前の動作コマンドデータの上位4ビットになります)。
4. デバイス → コントローラ  
5バイト目の送信データはSUM値の上位データ、6バイト目の送信データはSUM値の下位データになります。SUMの計算方法は、3.3.4.15「SUMの計算方法」を参照してください。
5. デバイス → コントローラ  
7バイト目の送信データは、CHECK SUMデータになります。5バイト目から6バイト目の送信データを符号なし8ビット加算(オーバーフローを無視)を行い、得られた下位8ビット値の2の補数を送信します。
6. コントローラ → デバイス  
8バイト目の受信データは、次の動作コマンドデータになります。

## 3.3.4.10 製品情報読み出しコマンド (表 3.3.10、表 3.3.11参照)

1. 1バイト目～2バイト目までの送受信データはRAM転送コマンドの場合と同一になります。
2. コントローラ → デバイス  
3バイト目の受信データは、動作コマンドデータになります。この場合は、製品情報読み出しコマンドデータ (30H) になります。
3. デバイス → コントローラ  
4バイト目の送信データは、3バイト目の動作コマンドデータに対するACK応答データになります。  
最初に、3バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常のACK応答データ (bit 3) x8Hを送信して、次の動作コマンド (3バイト目) データ待ちになります。送信データの上位4ビットは不定値になります(直前の動作コマンドデータの上位4ビットになります)。  
次に、3バイト目の受信データが、表3.3.7の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信(正常ACK応答データ)します。この場合、30Hをエコーバック送信して、製品情報読み出し処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーのACK応答データ (bit 0) x1Hを送信して、次の動作コマンド (3バイト目) データ待ちになります。送信データの上位4ビットは不定値になります(直前の動作コマンドデータの上位4ビットになります)。
4. デバイス → コントローラ  
5バイト目～8バイト目の送信データは、フラッシュメモリのデータ (02FEF0H～02FEF3H番地のデータ) になります。この番地にソフトなどのID情報を書き込んでおくことにより、書き込んだソフトのバージョン管理をすることができます。(例えば0002Hなら、書き込んだソフトがバージョン2と管理する。)
5. デバイス → コントローラ  
9バイト目～20バイト目の送信データは製品名になります。9バイト目から、アスキーコードで、'TMP91FW40\_ \_ \_'を送信します。  
注) ' ': スペース
6. デバイス → コントローラ  
21バイト目～24バイト目の送信データはパスワード比較開始アドレスになります。21バイト目から、F4H、FEH、02H、00Hを送信します。
7. デバイス → コントローラ  
25バイト目～28バイト目の送信データはRAM開始アドレスになります。25バイト目から、00H、10H、00H、00Hを送信します。
8. デバイス → コントローラ  
29バイト目～32バイト目の送信データはRAM(ユーザ領域)終了アドレスになります。29バイト目から、FFH、1DH、00H、00Hを送信します。
9. デバイス → コントローラ  
33バイト目～36バイト目の送信データはRAM終了アドレスになります。33バイト目から、FFH、1FH、00H、00Hを送信します。

## 10. デバイス → コントローラ

37 バイト目 ~ 44 バイト目の送信データは、ダミーデータになります。

## 11. デバイス → コントローラ

45 バイト目 ~ 46 バイト目の送信データは、プロテクトの設定状態や、フラッシュメモリがセクタ分割されているかを各ビットに割り付けて送信します。

- 0 ビット目は、リードプロテクトの設定状態を示します。
  - “0”はリードプロテクトが設定されている
  - “1”はリードプロテクトが設定されていない
- 1 ビット目は、ライトプロテクトの設定状態を示します。
  - “0”はライトプロテクトが設定されている
  - “1”はライトプロテクトが設定されていない
- 2 ビット目は、フラッシュメモリが分割されているかを示します。
  - “0”は分割されている
  - “1”は分割されていない
- 3 ビット目から 15 ビット目までは “0”を送信します。

## 12. デバイス → コントローラ

47 バイト目 ~ 50 バイト目の送信データは、フラッシュメモリ開始アドレスになります。  
47 バイト目から、00H、00H、01H、00H を送信します。

## 13. デバイス → コントローラ

51 バイト目 ~ 54 バイト目の送信データは、フラッシュメモリ終了アドレスになります。  
51 バイト目から、FFH、FFH、02H、00H を送信します。

## 14. デバイス → コントローラ

55 バイト目 ~ 56 バイト目の送信データは、フラッシュメモリのセクタ分割数になります。  
55 バイト目から、20H、00H を送信します。

## 15. デバイス → コントローラ

57 バイト目 ~ 65 バイト目の送信データは、フラッシュメモリのセクタ情報になります。  
セクタ情報は、フラッシュメモリ開始アドレスから見たとき、同一セクタサイズが何セクタ続いているかを一単位とし、同一セクタサイズの先頭の開始アドレスとセクタサイズ (ハーフワード) およびセクタの個数で表します。なお、セクタサイズはワードで表します。  
57 バイト目 ~ 65 バイト目の送信データは、4K バイトのセクタ (Sector0~Sector31) を表します。  
送信データについては、表 3.3.10、表 3.3.11を参照してください。

## 16. デバイス → コントローラ

66 バイト目の送信データは、CHECK SUM データになります。5 バイト目から 65 バイト目の送信データを符号なし 8 ビット加算 (オーバーフローを無視) して、得られた下位 8 ビット値の 2 の補数を送信します。

## 17. コントローラ → デバイス

67 バイト目の受信データは、次の動作コマンドデータになります。

## 3.3.4.11 フラッシュメモリチップ消去コマンド (表 3.3.12参照)

1. 1バイト目~2バイト目までの送受信データはRAM転送コマンドの場合と同一になります。
2. コントローラ → デバイス
  - 3 バイト目の受信データは動作コマンドデータになります。この場合は、フラッシュメモリチップ消去コマンドデータ (40 H) になります。
3. デバイス → コントローラ
  - 4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。  
次に、3 バイト目の受信データが、表 3.3.7の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常ACK応答データ) します。この場合、40H をエコーバック送信します。該当しない場合は、動作コマンドエラーのACK応答データ (bit 0) x1Hを送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。
4. コントローラ → デバイス
  - 5 バイト目の受信データは消去イネーブルコマンドデータ(54H)になります。
5. デバイス → コントローラ
  - 6 バイト目の送信データは、5 バイト目の消去イネーブルコマンドデータに対する ACK 応答データになります。  
最初に、5 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。  
次に、5 バイト目の受信データが、消去イネーブルコマンドデータに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、54H をエコーバック送信して、フラッシュメモリチップ消去処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。
6. デバイス → コントローラ
  - 7 バイト目の送信データは消去が正常に終了したかどうかを示します。  
正常に終了した時は、終了コード(4FH)を返します。  
消去 Error が起きた場合は、エラーコード(4CH)を返します。
7. デバイス → コントローラ
  - 8 バイト目の送信データは ACK 応答データです。  
正常に終了した時は、消去 ACK 応答コード(5DH)を返します。  
消去 Error の場合は、消去 ErrorACK 応答コード (60H)を返します。
8. コントローラ → デバイス
  - 9 バイト目の受信データは、次の動作コマンドデータになります。

## 3.3.4.12 フラッシュメモリプロテクト設定コマンド (表 3.3.13参照)

1. 1バイト目~2バイト目までの送受信データはRAM転送コマンドの場合と同一になります。
2. コントローラ → デバイス  
3バイト目の受信データは、動作コマンドデータになります。この場合は、フラッシュメモリプロテクト設定コマンドデータ (60H) になります。
3. デバイス → コントローラ  
4バイト目の送信データは、3バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3バイト目) データ待ちになります。送信データの上位4ビットは、不定値になります(直前の動作コマンドデータの上位4ビットになります)。  
次に、3バイト目の受信データが、表 3.3.7の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常ACK応答データ) します。フラッシュメモリプロテクト設定コマンドを選択した場合には、60Hをエコーバック送信してフラッシュメモリプロテクト設定処理ルーチンに分岐します。  
このルーチンに分岐後、パスワードエリアのデータをチェックします。パスワードエリアのデータのチェック方法は、3.3.4.14「パスワードについて」を参照してください。  
3バイト目の受信データがいずれのコマンドにも該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、再び動作コマンド (3バイト目) データ待ちになります。送信データの上位4ビットは、不定値になります(直前の動作コマンドデータの上位4ビットになります)。
4. コントローラ → デバイス  
5バイト目~16バイト目の受信データは、パスワードデータ (12バイト) になります。5バイト目の受信データはフラッシュメモリの 02FEF4H のデータと照合し、6バイト目の受信データはフラッシュメモリの 02FEF5H のデータと照合します。同様に 16バイト目の受信データはフラッシュメモリの 02FEFFH のデータと照合します。
5. コントローラ → デバイス  
17バイト目の受信データは、CHECK SUMデータになります。5バイト目から 16バイト目の送信データを符号なしの8ビット加算 (オーバフローを無視) して得られた下位8ビット値の2の補数をコントローラから送信してください。CHECK SUMデータの計算方法は、3.3.4.16「CHECK SUMの計算方法」を参照してください。

#### 6. デバイス → コントローラ

18 バイト目の送信データは、5 バイト目～17 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。最初に、5 バイト目～17 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 68H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので、“6” になります。

次に、17 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、5 バイト目～17 バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 61H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

最後に、パスワードの照合結果をチェックします。5 バイト目～16 バイト目のパスワードデータの照合がすべて一致しない場合、パスワードエラーの ACK 応答データ (bit 0) 61H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

上記のチェックを終えて、すべて正常なら、正常 ACK 応答データ 60H を送信します。

#### 7. デバイス → コントローラ

19 バイト目の送信データはプロテクト設定が正常に終了したかどうかを示します。

正常に終了した時は、終了コード(6FH)を返します。

Error が起きた場合は、エラーコード(6CH)を返します。

#### 8. デバイス → コントローラ

20 バイト目の送信データは ACK 応答データです。

正常に終了した時は、ACK 応答コード(31H)を返します。

Error の場合は、ErrorACK 応答コード (34H)を返します。

#### 9. デバイス → コントローラ

21 バイト目の受信データは、次の動作コマンドデータになります。

## 3.3.4.13 ACK応答データ

ブートプログラムは処理状況を各種コードによってコントローラに送信します。表 3.3.14から表 3.3.19に各受信データに対するACK応答データを示します。ACK応答データの上位 4 ビットは、動作コマンドデータの上位 4 ビットになります。また 3 ビット目は受信エラーを表し、0 ビット目は動作コマンドエラー、CHECK SUMエラー、パスワードエラーの状態を表します。

表 3.3.14 シリアル動作モード判定データに対する ACK 応答データ

送信データ	送信データの意味
86H	UART での通信が可能と判定した。(注)

注) ボーレートの設定が不可能と判定したら、何も送信しないで動作を停止します。

表 3.3.15 動作コマンドデータに対する ACK 応答データ

送信データ	送信データの意味
x8H (注)	動作コマンドデータに受信エラーが発生した。
x6H (注)	プロテクトが設定されており受信コマンドの動作を中止した。
x1H (注)	未定義の動作コマンドデータを正常受信した。
10H	RAM 転送コマンドと判定した。
20H	フラッシュメモリ SUM コマンドと判定した。
30H	製品情報読み出しコマンドと判定した。
40H	フラッシュメモリチップ消去コマンドと判定した。
60H	フラッシュメモリプロテクト設定コマンドと判定した。

注) 上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになります。

表 3.3.16 RAM 転送コマンド中の CHECK SUM データに対する ACK 応答データ

送信データ	送信データの意味
18H	受信エラーが発生していた。
11H	CHECK SUM エラーが発生した。あるいは、パスワードエラーが発生した。
10H	CHECK SUM 値は正常な値と判定した。

表 3.3.17 フラッシュメモリチップ消去動作に対する ACK 対応データ

送信データ	送信データの意味
54H	消去イネーブルコマンドと判定した。
4FH	消去終了
4CH	消去エラーが発生した。
5DH(注)	消去の再確認
60H(注)	消去エラーの再確認

注) 再確認表記は通信の確認

表 3.3.18 フラッシュメモリプロテクト設定コマンド中の CHECK SUM データに対する ACK 応答データ

送信データ	送信データの意味
68H	受信エラーが発生していた。
61H	CHECK SUM エラーが発生した。あるいは、パスワードエラーが発生した。
60H	CHECK SUM 値は正常な値と判定した。

表 3.3.19 フラッシュメモリプロテクト設定動作に対する ACK 対応データ

送信データ	送信データの意味
6FH	プロテクト(リード/ライト)設定終了
6CH	プロテクト(リード/ライト)設定エラーが発生した。
31H(注)	プロテクト(リード/ライト)設定の再確認
34H(注)	プロテクト(リード/ライト)設定エラーの再確認

注) 再確認表記は通信の確認

## 3.3.4.14 パスワードについて

動作コマンドデータが RAM 転送コマンド (10H) とフラッシュメモリプロテクト設定コマンド (60H) の場合、パスワードのチェックを行います。まず、動作コマンドデータをエコーバック送信 (10H または 60H) 後、パスワードエリア (02FEF4 番地~02FEFF 番地) のデータ (12 バイト) をチェックします。

次に、5 バイト目 ~ 16 バイト目の受信データ (パスワードデータ) の照合を行います。表 3.3.20 に対応表を示します。

パスワードが 12 バイト分すべて一致しないと、パスワードエラーになります。

また、パスワードが 12 バイト連続して同一データの場合もパスワードエラーになります。但し、パスワードが全て“FFH”で一致し、且つリセットベクタ (02FF00 番地~02FF02 番地) が全て“FFH”の場合は、ブランク品と判断し、パスワードエラーにはなりません。

パスワードエラーと判定された場合、18 バイト目の CHECK SUM 値に対する ACK 応答は、パスワードエラーとなります。

表 3.3.20 受信データと照合するデータの対応

受信データ	照合するデータ
5 バイト目	02FEF4 番地のデータ
6 バイト目	02FEF5 番地のデータ
7 バイト目	02FEF6 番地のデータ
8 バイト目	02FEF7 番地のデータ
9 バイト目	02FEF8 番地のデータ
10 バイト目	02FEF9 番地のデータ
11 バイト目	02FEFA 番地のデータ
12 バイト目	02FEFB 番地のデータ
13 バイト目	02FEFC 番地のデータ
14 バイト目	02FEFD 番地のデータ
15 バイト目	02FEFE 番地のデータ
16 バイト目	02FEFF 番地のデータ

パスワードに指定できないデータ例

## ブランク品の場合(注)

- ALL“FF”(FFH, FFH, FFH) 以外のデータはパスワードに指定できません。ブランク品は必ず ALL“FF”を指定します。

注)ブランク品とはパスワードエリア (02FEF4 番地~02FEFF 番地) とリセットベクタ (02FF00 番地~02FF02 番地) が全て“FFH”となっている製品を示しています。

## 書き込み品の場合

- 12 バイト連続の同一データはパスワードに指定できません。下表に書き込み品のパスワードエラー例を示します。

書き込み品	1	2	3	4	5	6	7	8	9	10	11	12	備考
エラー例 1	FFH	ALL“FF”											
エラー例 2	00H	ALL“00”											
エラー例 3	5AH	ALL“5A”											

### 3.3.4.15 SUMの計算方法

SUM の計算方法は、バイト+バイト+バイト+……+バイトの結果をワードで返します。つまり、バイトでデータを読み出して符号なし 8 ビット加算を行い、計算結果を 16 ビットで求めています。

コントローラへは、SUM の上位 8 ビットデータ、下位 8 ビットデータの順番で送信します。SUM の計算対象のデータは、フラッシュメモリ全エリア (128K バイト) のデータになります。フラッシュメモリ SUM コマンドを実行したときに返される SUM は、本計算方法を使用しています。

例)	A1H	左記 4 バイトが計算対象データの場合、SUM の値は、 $A1H + B2H + C3H + D4H = 02EAH$ となるので、 SUM の上位のデータは、02H SUM の下位のデータは、EAH になります。 したがって、コントローラには 02H、EAH の順番で 送信します。
	B2H	
	C3H	
	D4H	

### 3.3.4.16 CHECK SUMの計算方法

CHECK SUM の計算方法は、送信データを符号なし 8 ビット加算 (オーバフローを無視)して得られた下位 8 ビット値の 2 の補数値を求めています。フラッシュメモリ SUM コマンド、製品情報読み出しコマンドを実行したときに返される CHECK SUM は、本計算方法を使用しています。また、コントローラは CHECK SUM 値を送信するときは、本計算方法を使用してください。

例) フラッシュメモリ SUM コマンドのときを例に説明します。

SUM の上位 8 ビットデータが E5H、下位 8 ビットデータが F6H の場合の CHECK SUM 値を求めます。

まず、符号なし 8 ビット加算して得られた値を求めます。

$$E5H + F6H = 1DBH$$

この値の下位 8 ビットに対しての 2 の補数をとると以下のようになり、この値が CHECK SUM 値になります。したがって、コントローラには 25H を送信します。

$$0 - DBH = 25H$$

### 3.3.5 ユーザブートモード (シングルチップモード上)

ユーザブートモードでは、ユーザ作成のフラッシュメモリ書き込み/消去プログラムを使用してフラッシュメモリを書き替えることが出来ます。シングルチップモードにおいて、通常のユーザアプリケーションプログラムが動作しているノーマルモードから、フラッシュメモリを書き替えるためのユーザブートモードに移行させることでフラッシュメモリの書き替えが行えます。

例えば、シングルチップモード起動時にノーマル/ユーザブートモードを確定したい場合は、条件判定を行うプログラムをユーザアプリケーションプログラムのリセット処理ルーチンの中に組み込みます。なお、モード切り替えの条件設定は、本デバイスの I/O を使用してユーザのボード条件に合わせて独自に構築してください。

従って、ユーザブートモードにてフラッシュメモリの書き替えを行う場合には、あらかじめフラッシュメモリ書き込み/消去プログラムをユーザアプリケーションプログラムに組み込んでおいてください。なお、内蔵フラッシュメモリは消去/書き込み動作中はフラッシュメモリのデータを読み出せないため、フラッシュメモリ書き込み/消去プログラムはフラッシュメモリ外に格納して実行させる必要があります。また、ユーザブートモードで内蔵フラッシュメモリの消去/書き込み動作中は、割り込み発生を禁止してください。

書き込み/消去ルーチンを内蔵フラッシュメモリに置く場合と、外部から転送する場合の 2 ケース を例に、以下(1-A)、(1-B) にその手順を説明します。

Not Recommended for New Design

## 3.3.5.1 (1-A)書き込み／消去方法手順例1

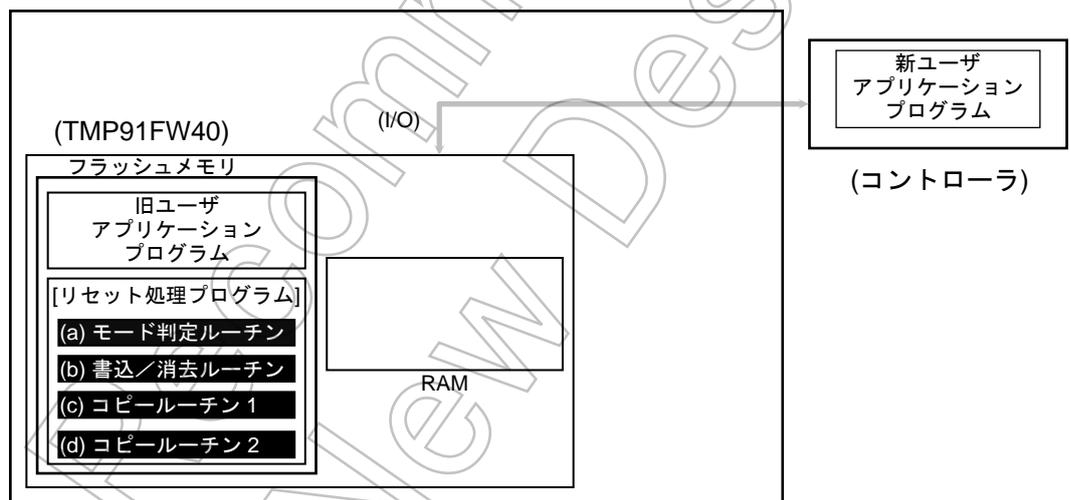
書き込み／消去プログラムをフラッシュメモリにあらかじめ内蔵している場合の手順例

## (Step-1)環境準備

ユーザは、あらかじめどのような条件(例えば端子状態)に設定されたらユーザブート動作に移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合ったデバイス周辺回路の設計、プログラムの作成を行います。ユーザは、あらかじめフラッシュメモリ上の任意のブロックに、以下に示す4つのプログラムを書き込んでおきます。

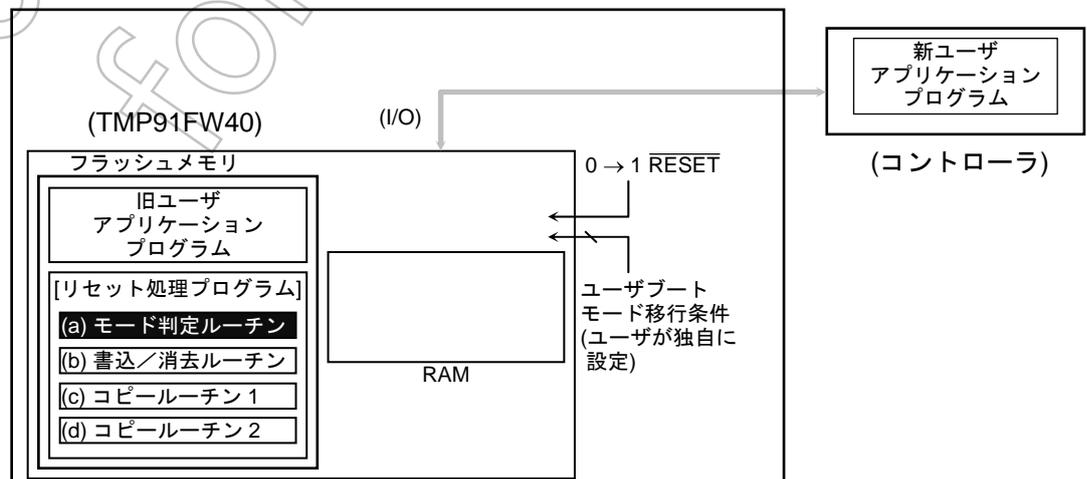
- (a) モード判定ルーチン : 書き込み／消去動作に移るためのプログラム
- (b) フラッシュ書き替えルーチン : 書き込み／消去データを外部から取り込み、フラッシュメモリを書き込み／消去するためのプログラム
- (c) コピー処理プログラム 1 : (a)~(d) を内蔵 RAM または外部メモリにコピーするためのプログラム
- (d) コピー処理プログラム 2 : 内蔵 RAM または外部メモリに書かれた(a)~(d)をフラッシュメモリへコピーするためのプログラム

注)(d)は書き替えプログラムをフラッシュ上に復元するプログラムです。常にフラッシュメモリの全領域が書き替えになり、書き替えデータのプログラムに同様のプログラムが存在する場合は不要になります。



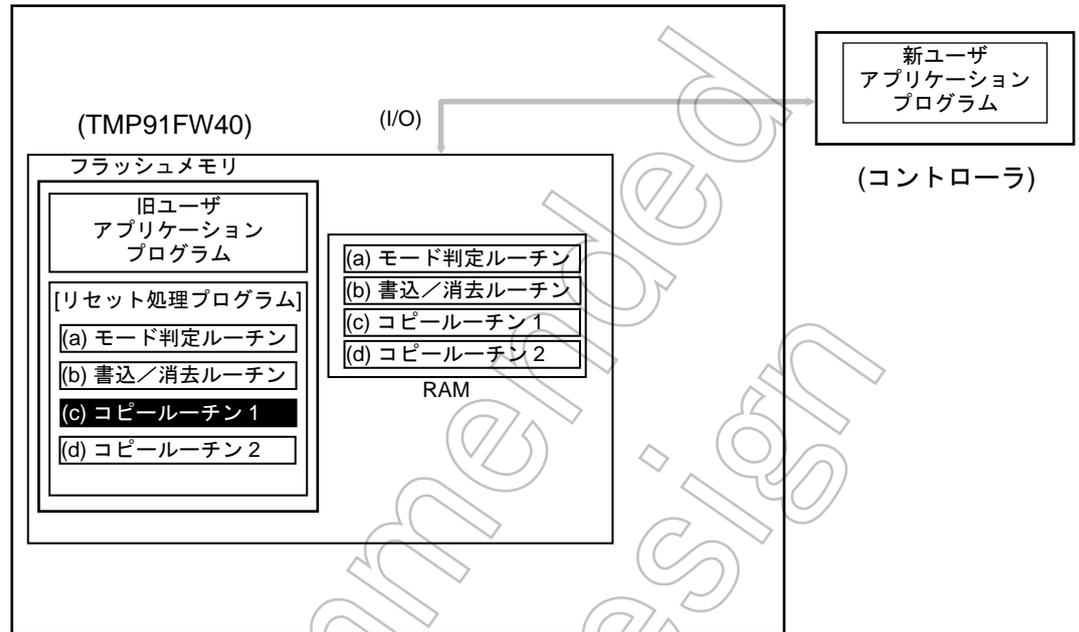
## (Step-2) ブートの起動(モード判定を RESET 処理に組み込んだ場合)

リセット解除後のリセット処理プログラムにおいてユーザブートモードへの移行を判定します。このとき、移行条件(端子設定等)が整っていれば、プログラムは書き替えのためのユーザブートモードに移ります。



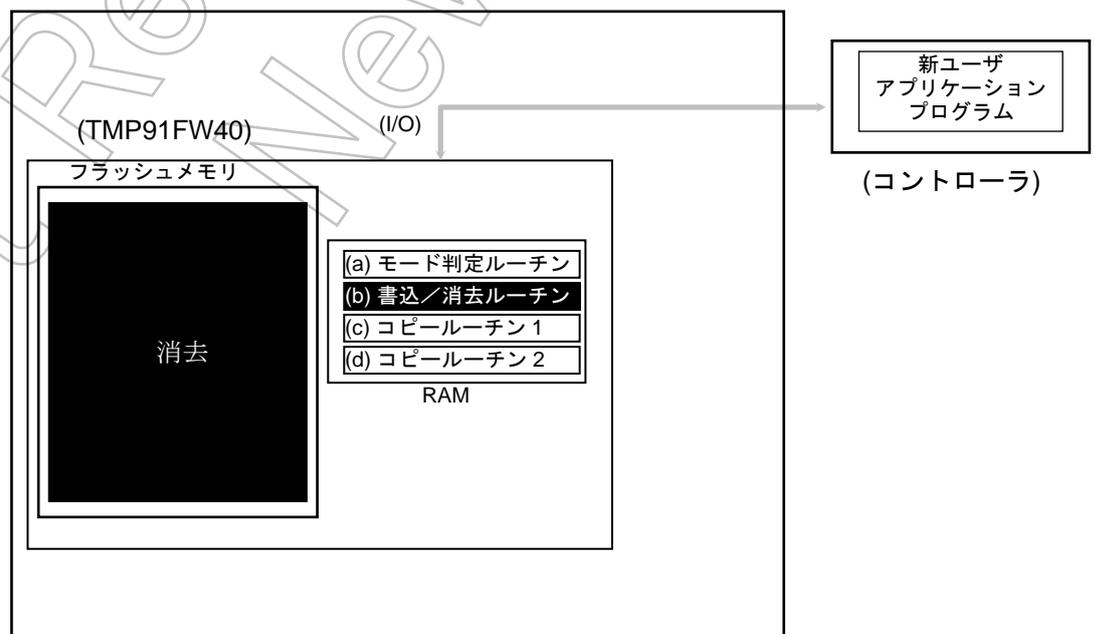
**(Step-3) ユーザブートの起動&RAM への書き込み/消去ルーチンのコピー**

ユーザブートモードに移ると、(c) コピールーチン 1 を使用して、(a)~(d) を内蔵 RAM もしくは外部メモリにコピーします。(下図は内蔵 RAM へコピーした場合を示します。)

**(Step-4) 書き換えルーチンによるフラッシュの消去**

RAM 上の書き込み/消去ルーチンへジャンプし、旧ユーザプログラムエリアの消去(セクタ単位、もしくはチップ消去)を行います。(RAM 上からフラッシュメモリに消去コマンドを与える場合。)

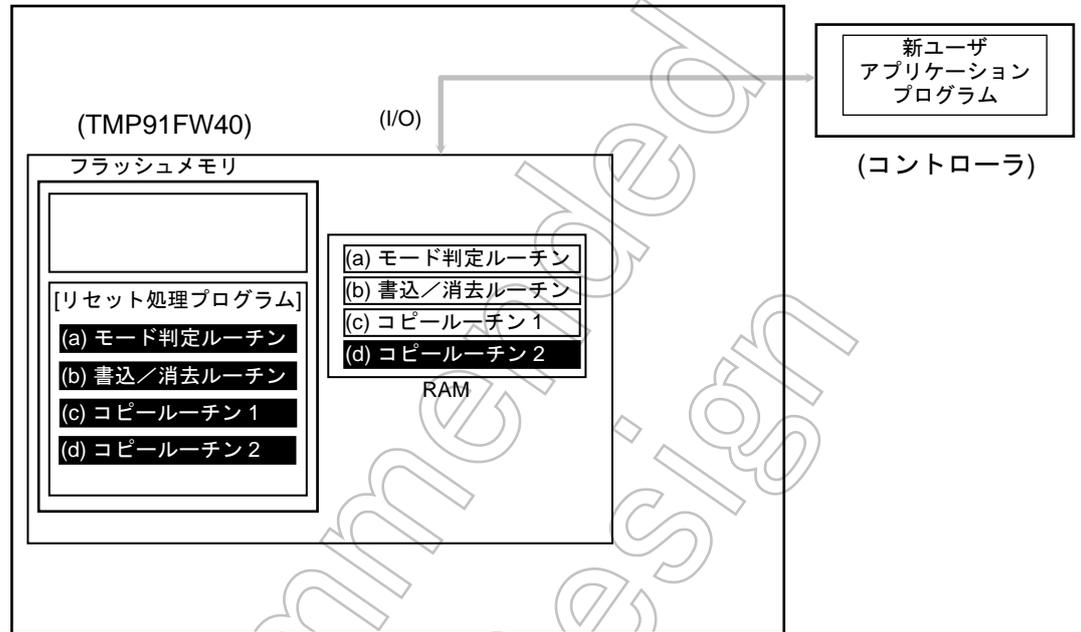
注)セクタ単位消去により、(a)~(d)のルーチンをフラッシュメモリに残す場合は(b)の書き込み/消去ルーチンだけ、RAM 上にコピーします。



(Step-5) ユーザブートプログラムのフラッシュメモリ上への復帰

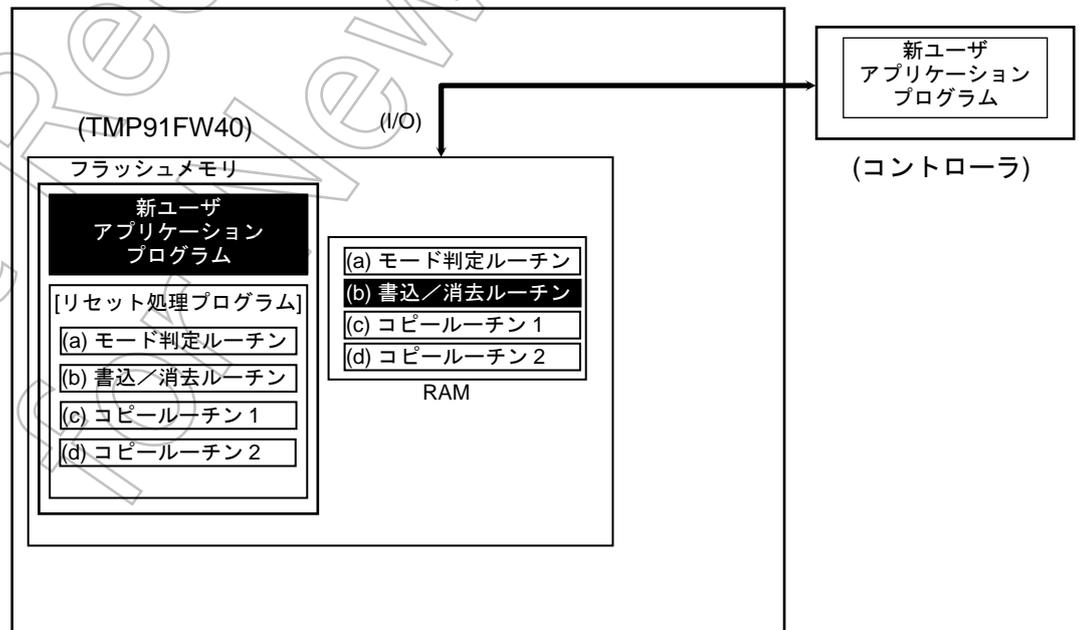
RAM 上の コピールーチン 2 を使用して、(a)~(d) をフラッシュメモリにコピーします。

注)セクタ単位消去により、(a)~(d)のルーチンをフラッシュメモリに残す場合、Step5 は不要です。



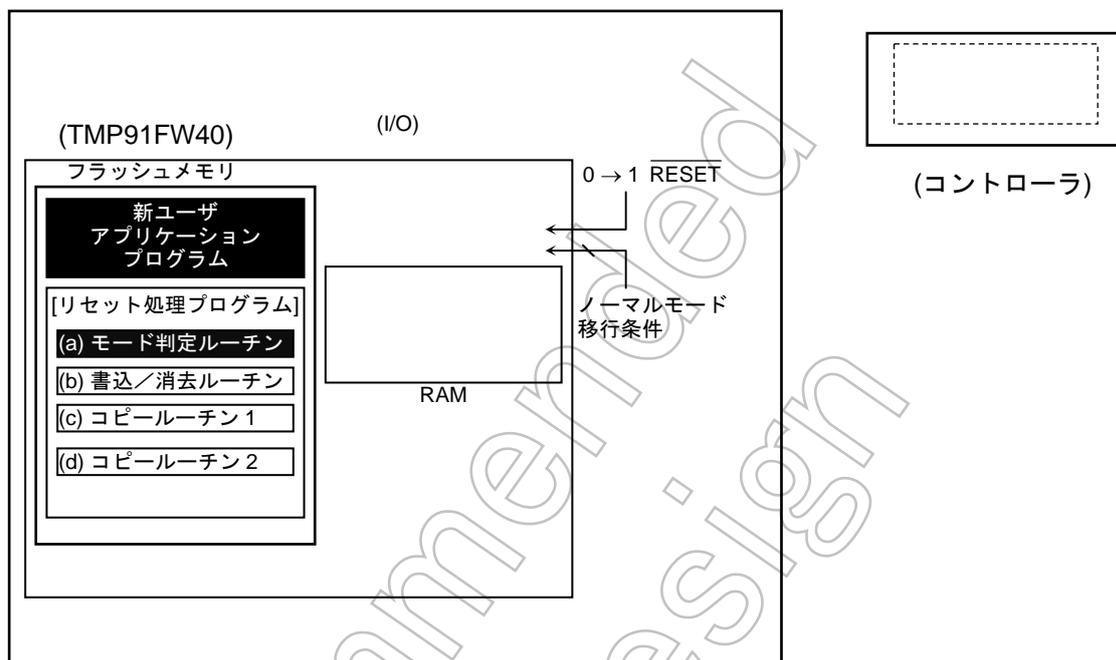
(Step-6) 新ユーザアプリケーションプログラムのフラッシュへの書き込み

さらに、RAM 上の書き替えルーチンを実行して、転送元 (コントローラ) より新ユーザアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。



(Step-7) 新ユーザアプリケーションプログラムの起動

RESET入力端子を“0”にしてリセットを行い、設定条件をノーマルモードに設定します。リセット解除後、新ユーザアプリケーションプログラムで動作を開始します。



Not Recommended for New Design

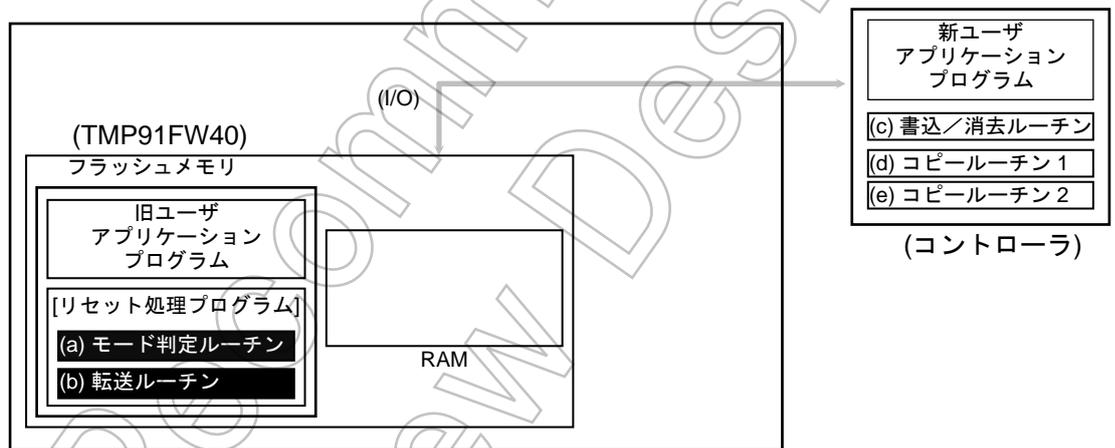
## 3.3.5.2 (1-B)書き込み／消去方法手順例2

以下は(1-A)とは異なり、必要最低限のブートプログラムをフラッシュ上に載せ、他に必要なプログラムはコントローラから供給する場合の手順です。

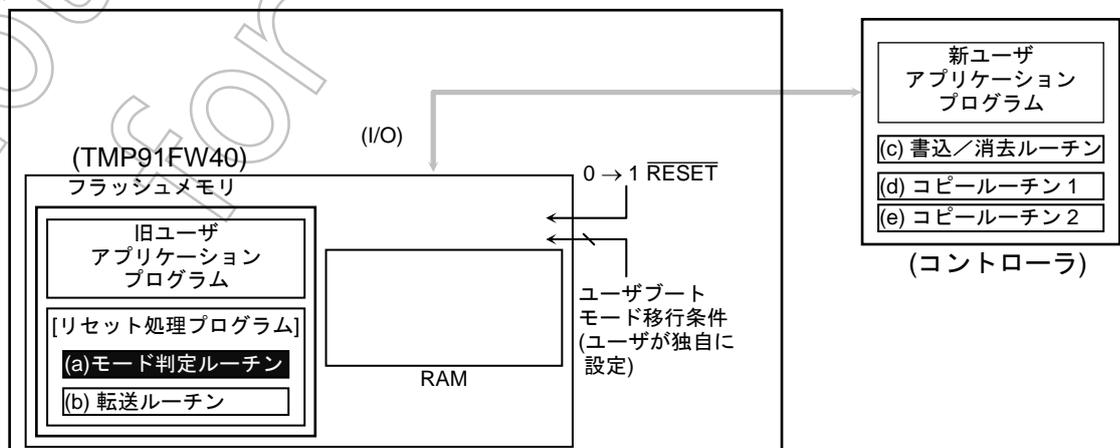
## (Step-1)環境準備

ユーザは、あらかじめどのような条件(例えば端子状態)に設定されたらユーザブート動作に移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合ったデバイス周辺回路の設計、プログラムの作成を行います。ユーザは、あらかじめフラッシュメモリ上の任意のブロックに、以下に示す2つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン : 書き込み／消去動作に移るためのプログラム
- (b) 転送ルーチン : 書き込み／消去プログラムを外部から取り込むためのプログラムまた、下記に示すルーチンはコントローラ上に用意します。
- (c) 書き込み／消去ルーチン : 書き込み／消去を行うためのプログラム
- (d) コピールーチン 1 : (a),(b)を内蔵 RAM または外部メモリにコピーするためのプログラム
- (e) コピールーチン 2 : 内蔵 RAM または外部メモリに書かれた(a),(b)をフラッシュメモリへコピーするためのプログラム

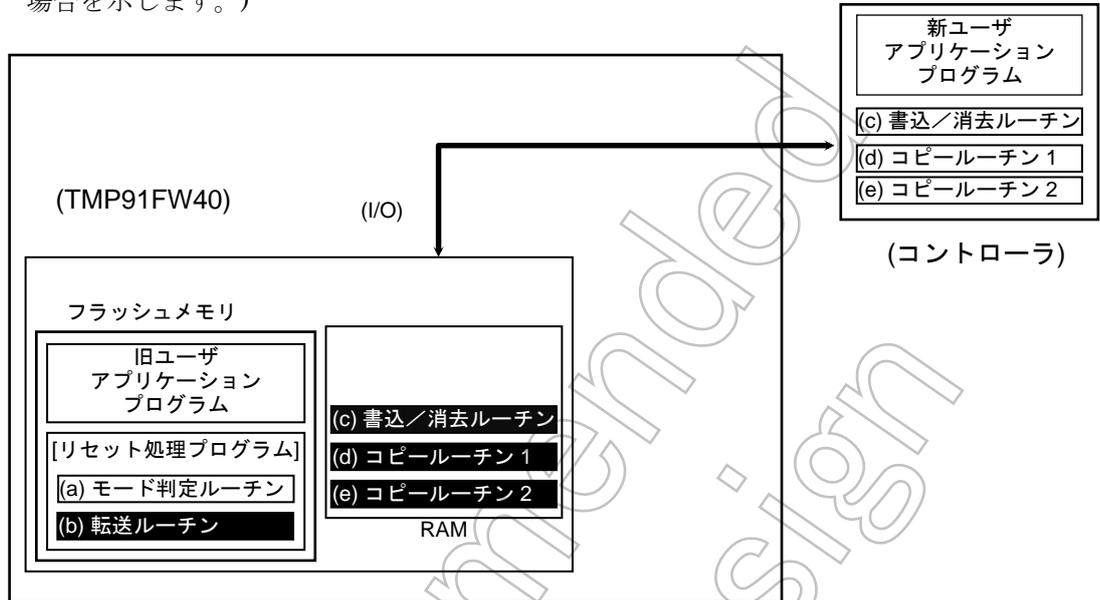
(Step-2) ブートの起動(モード判定を **RESET** 処理に組み込んだ場合)

リセット解除後のリセット処理プログラムにおいてユーザブートモードへの移行を判定します。このとき、移行条件(端子設定等)が整っていれば、プログラムは書き替えのためのユーザブートモードに移ります。



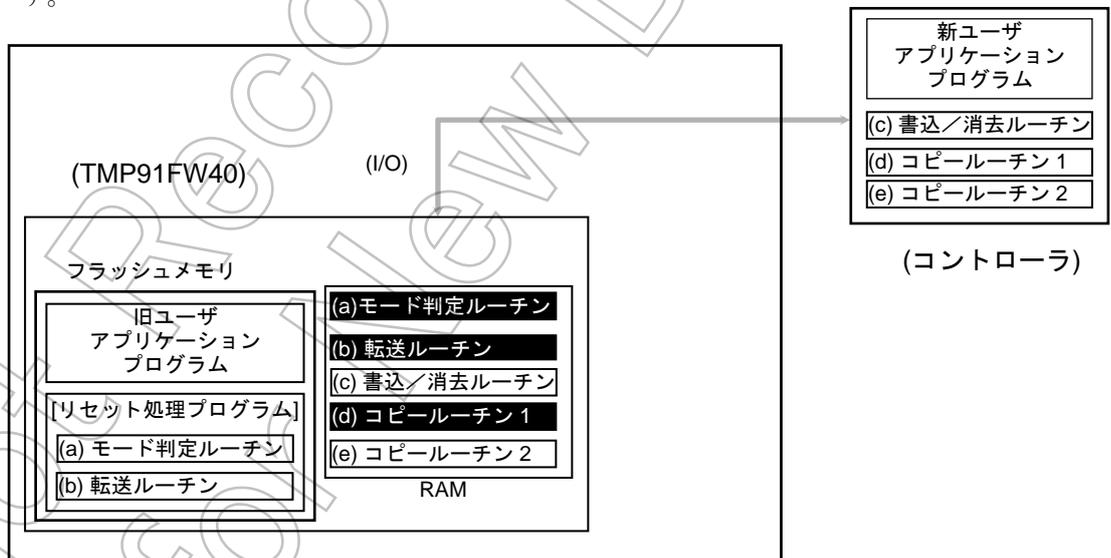
(Step-3) ユーザブートの起動&RAM への書き込み/消去ルーチンのコピー

ユーザブートモードに移ると、(b) 転送ルーチンを使用して、転送元(コントローラ)より (c)~(e)を内蔵 RAM (又は外部メモリ)にロードします。(下図は内蔵 RAM へコピーした場合を示します。)



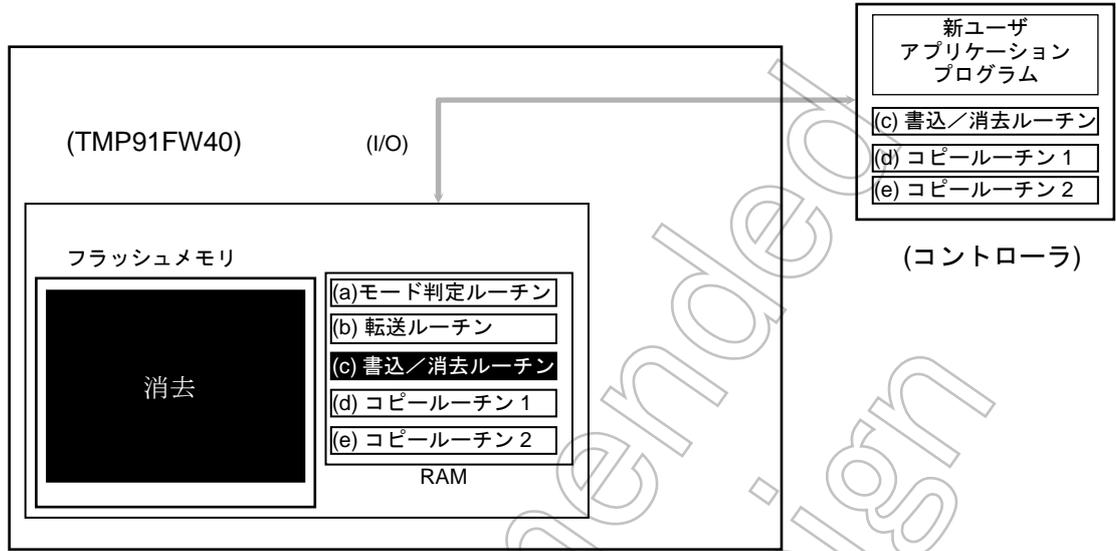
(Step-4) RAM 上のルーチンを起動

RAM 上に制御を移し、(d) コピールーチン 1 を使用して、(a),(b) を RAM にコピーします。



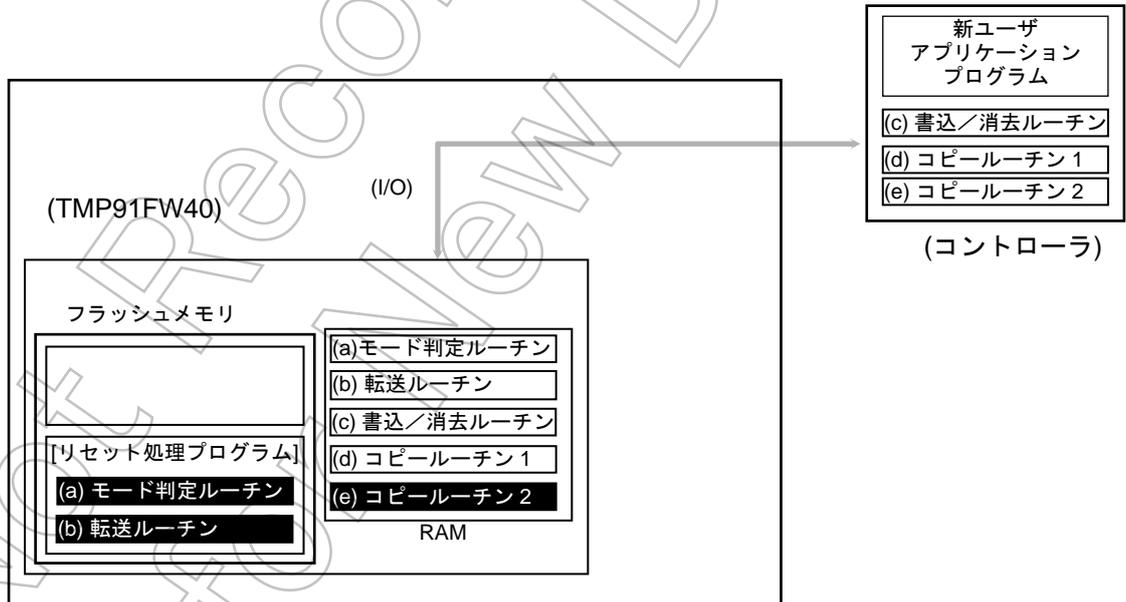
(Step-5) 書き替えルーチンによるフラッシュの消去

(c)の書き替えルーチンを使用して、旧ユーザプログラムエリアの消去を行います。



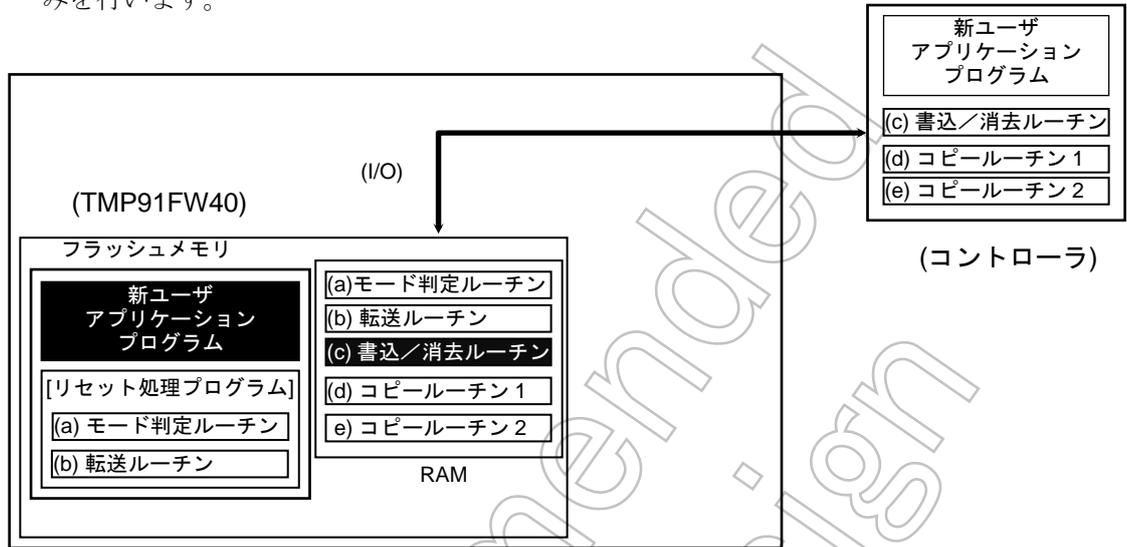
(Step-6) ユーザブートプログラムのフラッシュメモリ上への復帰

(e)のコピールーチン2を使用して、RAM上の(a),(b)をフラッシュメモリにコピーします。



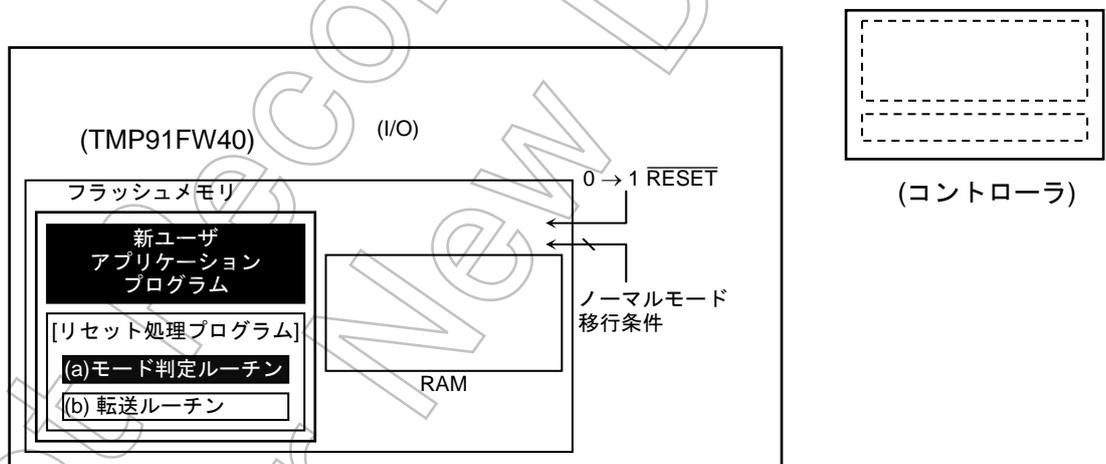
(Step-7) 新ユーザアプリケーションプログラムのフラッシュへの書き込み

さらに、RAM 上の (c) 書き込み/消去ルーチンを実行して、転送元 (コントローラ) より新ユーザアプリケーションプログラムのデータをロードし、消去したエリアに書き込みを行います。



(Step-8) 新ユーザアプリケーションプログラムの起動

RESET 入力端子を“0”にしてリセットを行い、設定条件をノーマルモードに設定します。リセット解除後、新ユーザアプリケーションプログラムで動作を開始します。



### 3.3.6 フラッシュメモリコマンドシーケンス

フラッシュメモリの動作は、6つのコマンドから構成されています。表 3.3.21にコマンドシーケンスの詳細を示します。コマンドシーケンスで指定するアドレスは、フラッシュメモリがマッピングされている領域の何れかを指定する必要があります。詳細は、表 3.3.3を参照してください。

表 3.3.21 コマンドシーケンス

	コマンド シーケンス	第1バス ライトサイクル		第2バス ライトサイクル		第3バス ライトサイクル		第4バス ライトサイクル		第5バス ライトサイクル		第6バス ライトサイクル	
		Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data	Addr.	Data
1	1ワード書き込み	AAAH	AAH	554H	55H	AAAH	A0H	PA (注1)	PD (注1)				
2	セクタイレース (4KB単位の部分消去)	AAAH	AAH	554H	55H	AAAH	80H	AAAH	AAH	554H	55H	SA (注2)	30H
3	チップイレース (全面消去)	AAAH	AAH	554H	55H	AAAH	80H	AAAH	AAH	554H	55H	AAAH	10H
4	Product ID Entry	AAAH	AAH	554H	55H	AAAH	90H						
5	Product ID Exit	xxH	F0H										
	Product ID Exit	AAAH	AAH	554H	55H	AAAH	F0H						
6	リードプロテクト設定	AAAH	AAH	554H	55H	AAAH	A5H	77EH	F0H (注3)				
	ライトプロテクト設定	AAAH	AAH	554H	55H	AAAH	A5H	77EH	0FH (注3)				

注1) PA=プログラムワードアドレス、PD=プログラムワードデータ

書き込みを行うアドレスとデータを設定してください。アドレスは偶数アドレスを設定してください。

注2) SA=セクタアドレス

アドレスの A23~A12 で個々のセクタ消去範囲が選択されます。

注3) リードプロテクトとライトプロテクトの両方を設定する場合は、データ 00H を書き込み必ず一度に設定してください。

表 3.3.22 ハードウェアシーケンスフラグー覧

状態		D7	D6
自動動作実行中	1ワード書き込み	D7反転	トグル
	セクタ/チップイレース	0	トグル
	リード/ライトプロテクト設定	使用不可	トグル

注) D15~D8、D5~D0はDon't care。

### 3.3.6.1 1ワード書き込み

1ワード単位でフラッシュメモリの書き込みを行います。第4バスライトサイクルで書き込みを行うアドレスとデータを指定します。1ワードあたりの書き込み時間は最大60 $\mu$ sです。書き込みが終了するまでは、他のコマンドシーケンスを実行することができません。書き込み終了を確認するには、フラッシュメモリの同一アドレスをリードし同一データが読み出せるまでポーリングを行います。書き込み中はリードするたびにデータビット6が反転します。

注) 既にデータ(FFFFHを含む)が書き込まれたフラッシュメモリのアドレスに対して、再度データの書き込みを行う場合は、セクタイレースまたはチップイレースによって、必ずそのアドレスのデータを消去した後に書き込みを実行してください。

### 3.3.6.2 セクタイレース(4KB単位の部分消去)

4Kバイト単位でフラッシュメモリの消去を行います。消去範囲は第6バスライトサイクルのアドレスで指定します。アドレスの指定は表3.3.3を参照してください。なお、セクタイレースは、ライターモード時は動作しません。

4Kバイトあたりの消去時間は、最大75msです。消去が終了するまでは、他のコマンドシーケンスを実行することができません。消去終了を確認するには、フラッシュメモリの同一アドレスをリードし同一データが読み出せるまでポーリングを行います。消去中はリードするたびにデータビット6が反転します。

### 3.3.6.3 チップイレース(全面消去)

フラッシュメモリの全領域を消去します。

全領域の消去時間は、最大300msです。消去が終了するまでは、他のコマンドシーケンスを実行することができません。消去終了を確認するには、フラッシュメモリの同一アドレスをリードし同一データが読み出せるまでポーリングを行います。消去中はリードするたびにデータビット6が反転します。

なお、消去された領域のデータはFFHとなります。

### 3.3.6.4 Product ID Entry

Product ID Entryを実行するとProduct IDモードが起動します。Product IDモード中、フラッシュメモリに対してリード命令を実行するとベンダーID、フラッシュマクロID、フラッシュサイズID、リード/ライトプロテクトステータスを読み出すことができます。なお、Product IDモード中は、フラッシュメモリのデータは読み出せません。

### 3.3.6.5 Product ID Exit

Product IDモードを終了します。

### 3.3.6.6 リードプロテクト設定

フラッシュメモリに対してリードプロテクトを設定します。リードプロテクトを設定するとライターモードのときフラッシュメモリのリードが出来なくなります。シングルブートモードのときは、RAM 転送コマンドが実行できなくなります。

リードプロテクト設定を解除するには、チップイレースを実行する必要があります。リードプロテクトが設定されているか確認するには Product ID モードで xxx77EH をリードします。リードプロテクトの設定時間は最大 60 $\mu$ s です。リードプロテクトの設定が終了するまでは、他のコマンドシーケンスを実行することができません。リードプロテクト設定の終了を確認するには、フラッシュメモリの同一アドレスをリードし同一データが読み出せるまでポーリングを行います。リードプロテクト設定中はリードするたびにデータビット 6 が反転します。

### 3.3.6.7 ライトプロテクト設定

フラッシュメモリに対してライトプロテクトを設定します。ライトプロテクトを設定するとライターモードのときフラッシュメモリのライトが出来なくなります。シングルブートモードのときは、RAM 転送コマンドが実行できなくなります。

ライトプロテクト設定を解除するには、チップイレースを実行する必要があります。ライトプロテクトが設定されているか確認するには Product ID モードで xxx77EH をリードします。ライトプロテクトの設定時間は最大 60 $\mu$ s です。ライトプロテクトの設定が終了するまでは、他のコマンドシーケンスを実行することができません。ライトプロテクト設定の終了を確認するには、フラッシュメモリの同一アドレスをリードし同一データが読み出せるまでポーリングを行います。ライトプロテクト設定中はリードするたびにデータビット 6 が反転します。

### 3.3.6.8 ハードウェアシーケンスフラグ

フラッシュメモリの自動動作実行状態を、ハードウェアシーケンスフラグにより確認できます。

#### 1) DATA ポーリング(D7)

フラッシュメモリの書き込みを実行すると、書き込み処理が完了するまでの間、D7 に書き込んだデータの反転データを出力し、完了後は D7 のセルデータを出力します。D7 を読み出すことで動作状態の識別ができます。セクタ/チップイレース処理実行中は D7 から“0”を出力し、完了後は“1”(セルデータ)を出力します。その後 1 $\mu$ s 待つて読み出すと全ビットの書き込みデータが読み出せます。

リード/ライトプロテクト設定を実行した場合は、DATA ポーリング機能は使用できませんので、トグルビット(D6)にて動作状態を識別してください。

## 2) トグルビット(D6)

フラッシュメモリの書き込み、セクタ/チップイレース、リード/ライトプロテクト設定を実行すると、これらの処理が完了するまでの間、リードオペレーションによって読み出されるデータの6ビット目(D6)の値はリードするたびに0と1が交互に出力されます。これを利用すると各処理の完了をソフト的に確認することができます。通常はフラッシュメモリの同一アドレスに対しリードを行い同一データが読み込まれるまでポーリングを行います。なお、トグルビットの最初の値はかならず"1"になります。

注) 内蔵されているフラッシュメモリには、内部タイマ超過ビット(D5)機能が無いため、タイマにて DATA ポーリングおよびトグルビットのポーリング時間を設定して、書き込み動作、イレース動作の最大時間を超えた場合にポーリングを中止するようにプログラムを設定する必要があります。

### 3.3.6.9 データリード

フラッシュメモリからのデータリードは、バイト単位またはワード単位で行います。なお、データリードの場合はコマンドシーケンスの実行は必要ありません。

Not Recommended for New Design

### 3.3.6.10 内部CPUによるフラッシュメモリ書き替え

内部 CPU によるフラッシュメモリ書き替えは、上述のコマンドシーケンス、ハードウェアシーケンスフラグを使って行います。ただし、内蔵フラッシュメモリは、自動動作モード中はメモリデータを読み出せないため、書き替えプログラムをフラッシュメモリエリア外で実行する必要があります。

内部 CPU によるフラッシュメモリ書き替えには、2通りの方法があります。あらかじめ用意されたシングルブートモードを使う方法と、シングルチップモード上でユーザ独自のプロトコルを使う方法(ユーザブート)です。

#### 1) シングルブート:

マイコンをシングルブートモードで起動させ、内蔵ブート ROM プログラムにより、フラッシュメモリを書き替える方法です。このモードでは、内蔵ブート ROM が割り込みベクタテーブルを含む領域にマッピングされ、ブート ROM プログラムが実行されます。また、フラッシュメモリはブート ROM 領域とは別のアドレス空間にマッピングされます。ブート ROM プログラムは、シリアル転送による書き替え用データ取り込み、およびフラッシュメモリの書き替えを行います。シングルブートは割り込み禁止状態で行います。また、ノンマスクابل割り込み(NMI等)も発生しないように処理しておく必要があります。詳細は 3.3.4「シングルブートモード」を参照してください。

#### 2) ユーザブート:

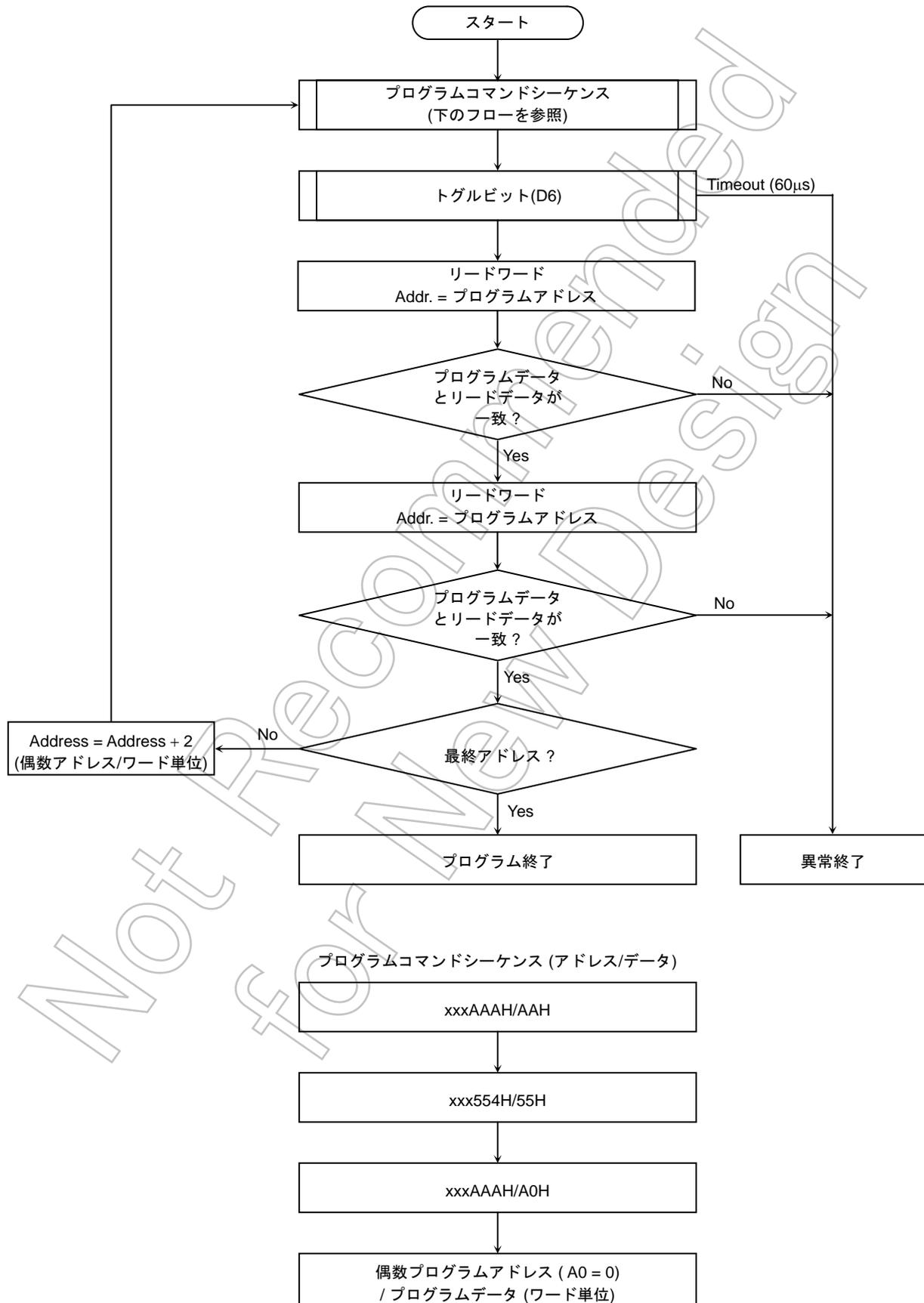
ユーザ独自のフラッシュメモリ書き替えプログラムを使う方法です。シングルチップモード(通常動作モード)で実行します。このモードでも、フラッシュメモリ領域とは別のアドレス空間上でフラッシュメモリ書き替えプログラムを実行させる必要があります。また、シングルブートと同様に、ノンマスクابلも含めたすべての割り込み発生を禁止する必要があります。

フラッシュメモリ書き替えプログラムは、書き替え用データ取り込みルーチン、フラッシュメモリ書き替えルーチンを含めて、あらかじめ用意しておきます。メインプログラム上で、通常動作からフラッシュメモリ書き替え動作へ切り替え、用意しておいたフラッシュメモリ書き替えプログラムをフラッシュメモリ領域外に展開して実行します。例えば、フラッシュメモリ書き替えプログラムをフラッシュメモリ上から内蔵 RAM へ展開して実行したり、外部メモリ上に用意して実行したりできます。

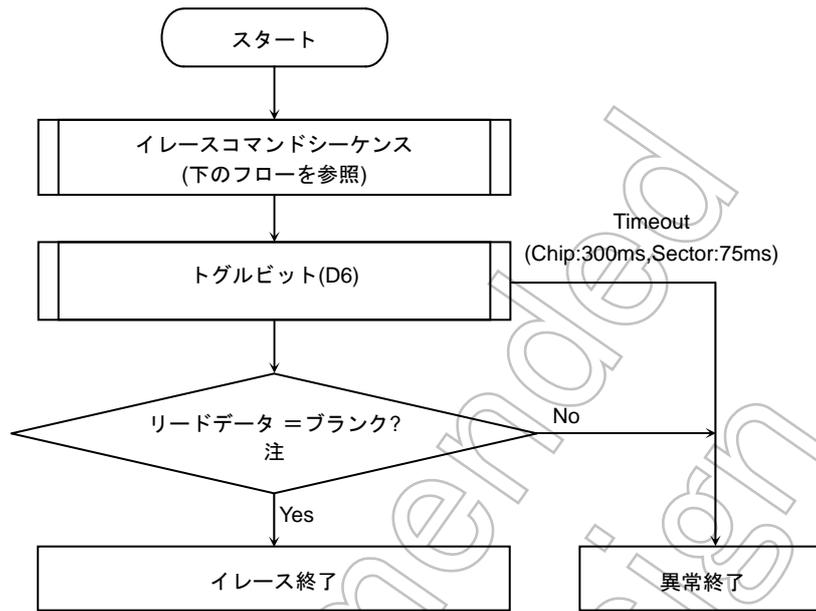
詳細は 3.3.5「ユーザブートモード(シングルチップモード上)」を参照してください。

フローチャート：内部 CPU によるフラッシュメモリアクセス

1 ワード書き込み

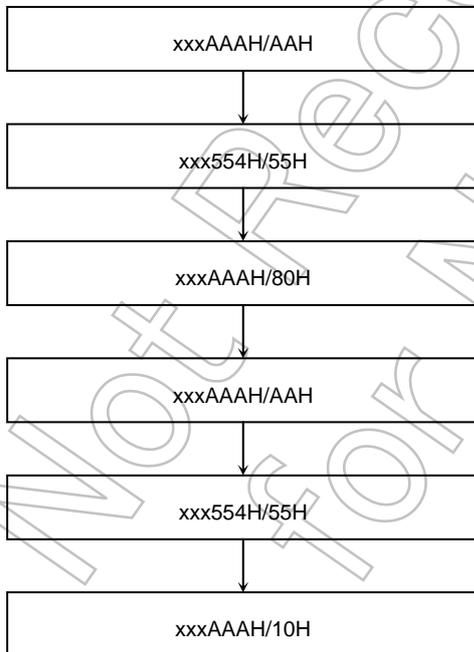


チップイレース/セクタイレース

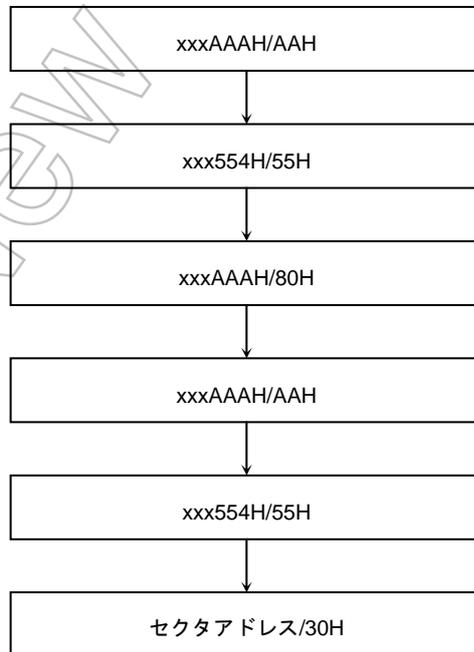


注) チップイレース時は、フラッシュメモリの全エリアがブランクであることを確認する。  
セクタイレース時は、選択したセクタエリアがブランクであることを確認する。

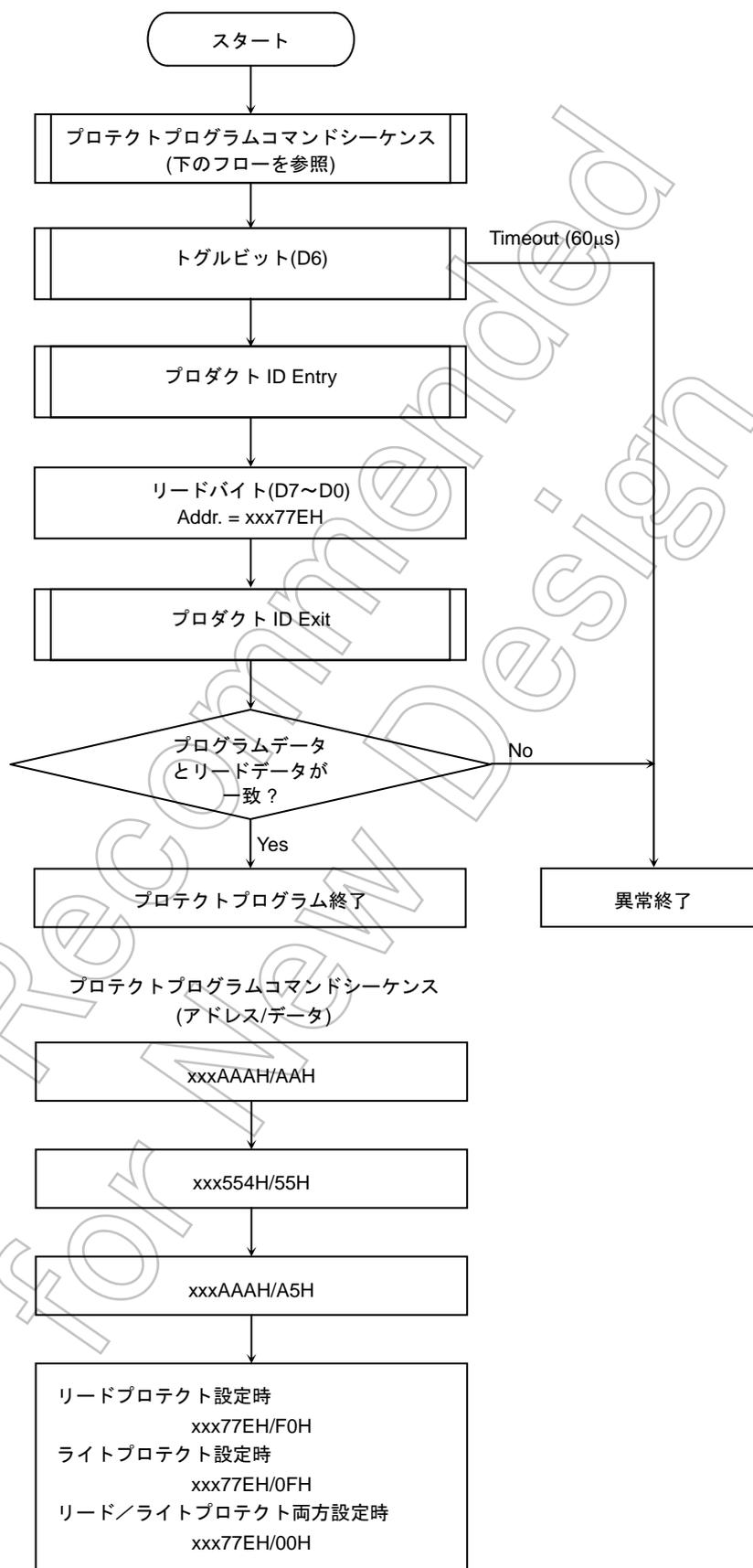
チップイレースコマンドシーケンス  
(アドレス/データ)

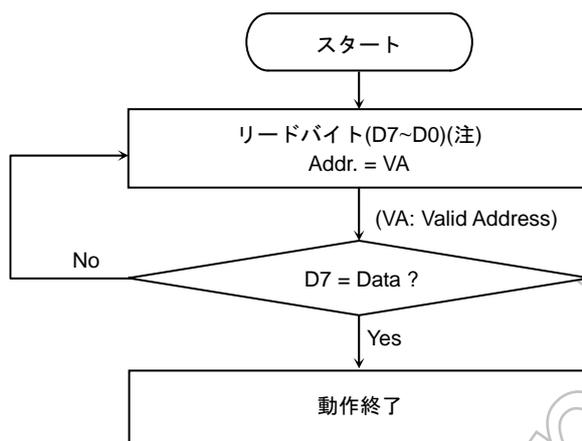
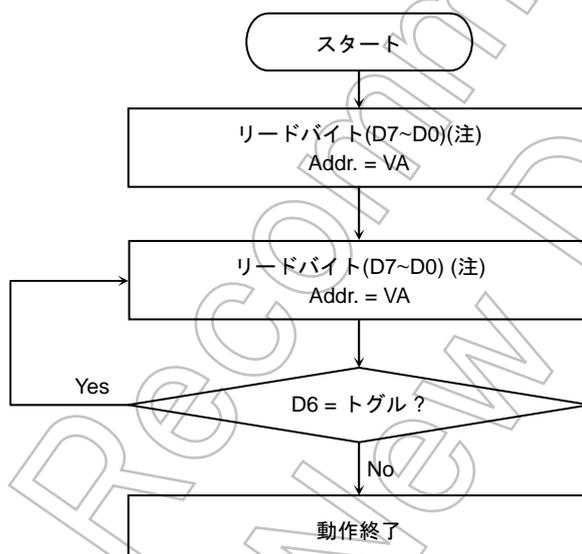


セクタイレースコマンドシーケンス  
(アドレス/データ)



## リード/ライトプロテクト設定



DATA ポーリング(D7)トグルビット(D6)

注) ハードウェアシーケンスフラグの読み出しは、バイト単位、またはワード単位で行います。

VA:1 ワード書き込み時は、書き込みを行っているアドレス。

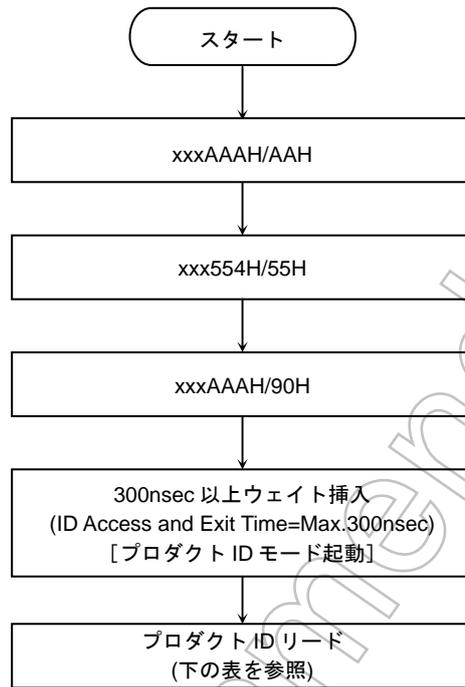
セクタイレース時は、選択したセクタ内の任意アドレス。

チップイレース時は、フラッシュメモリの任意アドレス。

リードプロテクト時は、プロテクト設定アドレス(0x77EH)。

ライトプロテクト時は、プロテクト設定アドレス(0x77EH)。

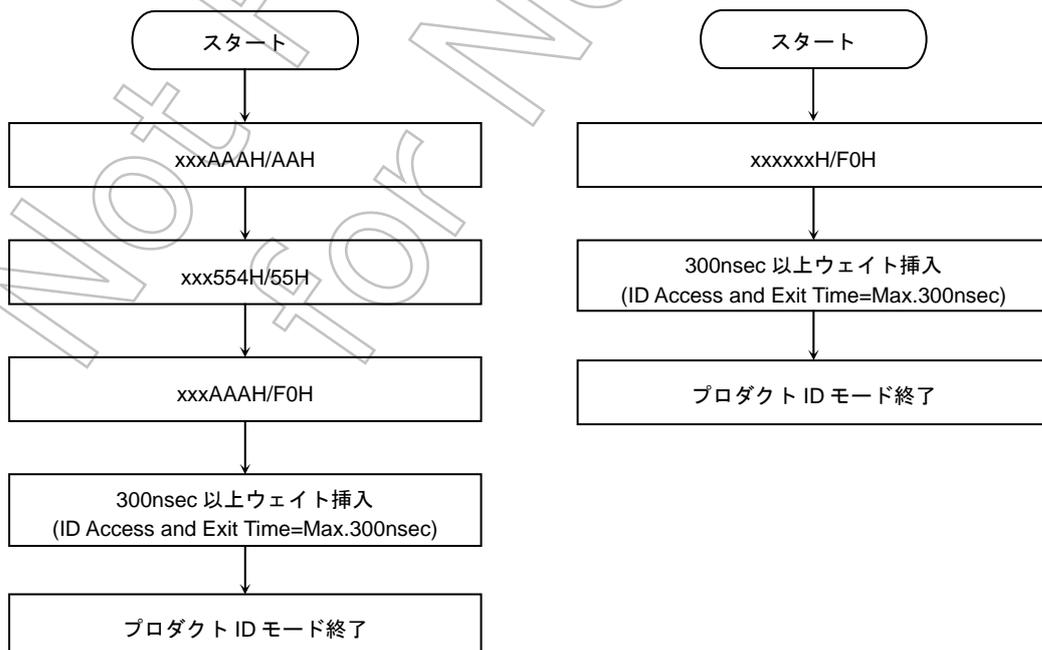
プロダクト ID Entry



プロダクト ID Entry 後のリード値

	アドレス	読み出される値
ベンダーID	xxx00H	98H
フラッシュマクロ ID	xxx02H	42H
フラッシュサイズ ID	xxx04H	1FH
リード/ライト プロテクトステータス	xx77EH	プロテクト設定時は書き込んだデータ プロテクト未設定時は FFH

プロダクト ID Exit



(プログラム例: RAMに展開するプログラム)

チップイレースを実行した後、FE000Hに0706Hのデータを書き込む

##### フラッシュメモリチップイレース処理 #####

```
ld XIX, 0xFE0000 ; スタートアドレスの設定
CHIPERASE:
ld (0xFE0AAA), 0xAA ; 1st Bus Write Cycle
ld (0xFE0554), 0x55 ; 2nd Bus Write Cycle
ld (0xFE0AAA), 0x80 ; 3rd Bus Write Cycle
ld (0xFE0AAA), 0xAA ; 4th Bus Write Cycle
ld (0xFE0554), 0x55 ; 5th Bus Write Cycle
ld (0xFE0AAA), 0x10 ; 6th Bus Write Cycle
```

```
cal TOGGLECHK ; トグルビットの確認
```

CHIPERASE\_LOOP:

```
ld WA, (XIX+) ; フラッシュメモリからデータ読み込み
cp WA, 0xFFFF ; ブランクデータか?
j ne, CHIPERASE_ERR ; ブランクデータ以外の場合はエラー処理へジャンプ
cp XIX, 0xFFFFF ; エンドアドレス(0xFFFFF)か?
j ULT, CHIPERASE_LOOP ; 全メモリエリア確認後にループ処理を終了
```

##### フラッシュメモリ書き込み処理 #####

```
ld XIX, 0xFE0000 ; 書き込みアドレスの設定
ld WA, 0x0706 ; 書き込みデータの設定
PROGRAM:
ld (0xFE0AAA), 0xAA ; 1st Bus Write Cycle
ld (0xFE0554), 0x55 ; 2nd Bus Write Cycle
ld (0xFE0AAA), 0xA0 ; 3rd Bus Write Cycle
ld (XIX), WA ; 4th Bus Write Cycle
```

```
cal TOGGLECHK ; トグルビットの確認
```

```
ld BC, (XIX) ; フラッシュメモリからデータ読み込み
cp WA, BC
j ne, PROGRAM_ERR ; 書き込んだデータが読み込めなければエラー
ld BC, (XIX) ; フラッシュメモリからデータ読み込み
cp WA, BC
j ne, PROGRAM_ERR ; 書き込んだデータが読み込めなければエラー
```

PROGRAM\_END:

```
j PROGRAM_END ; 書き込み完了
```

##### トグルビット(D6)確認処理 #####

TOGGLECHK:

```
ld L, (XIX) ; トグルビット(D6)の確認
and L, 0y01000000 ; 1回目のトグルビットデータを保存
ld H, L
```

TOGGLECHK1:

```
ld L, (XIX) ; トグルビット(D6)の確認
and L, 0y01000000 ; トグルビット(D6)の確認
cp L, H ; トグルビット=トグル?
j z, TOGGLECHK2 ; トグルビットが前回の状態と同じ場合は処理を終了
ld H, L ; 今回の状態を保存
j TOGGLECHK1 ; 再チェック
```

TOGGLECHK2:

```
ret
```

##### エラー処理 #####

CHIPERASE\_ERR:

```
j CHIPERASE_ERR ; チップイレースエラー
```

PROGRAM\_ERR:

```
j PROGRAM_ERR ; 書き込みエラー
```

(プログラム例：RAMに展開するプログラム)

FF0000H~FF0FFFHのセクタイレースを実行した後、FF0000Hに0706Hのデータを書き込む

##### フラッシュメモリセクタイレース処理 #####

```
ld XIX, 0xFF0000 ; スタートアドレスの設定
SECTORERASE:
ld (0xFE0AAA), 0xAA ; 1st Bus Write Cycle
ld (0xFE0554), 0x55 ; 2nd Bus Write Cycle
ld (0xFE0AAA), 0x80 ; 3rd Bus Write Cycle
ld (0xFE0AAA), 0xAA ; 4th Bus Write Cycle
ld (0xFE0554), 0x55 ; 5th Bus Write Cycle
ld (XIX), 0x30 ; 6th Bus Write Cycle

cal TOGGLECHK ; トグルビットの確認
```

SECTORERASE\_LOOP:

```
ld WA, (XIX+) ; フラッシュメモリからデータ読み込み
cp WA, 0xFFFF ; ブランクデータか?
j ne, SECTORERASE_ERR ; ブランクデータ以外の場合はエラー処理へジャンプ
cp XIX, 0xFF0FFF ; エンドアドレス(0xFF0FFF)か?
j ULT, SECTORERASE_LOOP ; 該当セクタエリア確認後にループ処理を終了
```

##### フラッシュメモリ書き込み処理 #####

```
ld XIX, 0xFF0000 ; 書き込みアドレスの設定
ld WA, 0x0706 ; 書き込みデータの設定
PROGRAM:
ld (0xFE0AAA), 0xAA ; 1st Bus Write Cycle
ld (0xFE0554), 0x55 ; 2nd Bus Write Cycle
ld (0xFE0AAA), 0xA0 ; 3rd Bus Write Cycle
ld (XIX), WA ; 4th Bus Write Cycle

cal TOGGLECHK ; トグルビットの確認

ld BC, (XIX) ; フラッシュメモリからデータ読み込み
cp WA, BC ;
j ne, PROGRAM_ERR ; 書き込んだデータが読み込めなければエラー
ld BC, (XIX) ; フラッシュメモリからデータ読み込み
cp WA, BC ;
j ne, PROGRAM_ERR ; 書き込んだデータが読み込めなければエラー

PROGRAM_END:
j PROGRAM_END ; 書き込み完了
```

##### トグルビット(D6)確認処理 #####

```
TOGGLECHK:
ld L, (XIX)
and L, 0y01000000 ; トグルビット(D6)の確認
ld H, L ; 1回目のトグルビットデータを保存

TOGGLECHK1:
ld L, (XIX)
and L, 0y01000000 ; トグルビット(D6)の確認
cp L, H ; トグルビット=トグル?
j z, TOGGLECHK2 ; トグルビットが前回の状態と同じ場合は処理を終了
ld H, L ; 今回の状態を保存
j TOGGLECHK1 ; 再チェック

TOGGLECHK2:
ret
```

##### エラー処理 #####

```
SECTORERASE_ERR:
j SECTORERASE_ERR ; セクタイレースエラー

PROGRAM_ERR:
j PROGRAM_ERR ; 書き込みエラー
```

(プログラム例 : RAM に展開するプログラム)

リードプロテクトとライトプロテクトの両方を設定する

##### フラッシュメモリプロテクト設定処理 #####

```
ld    XIX, 0xFE077E          ; プロテクトアドレスの設定
PROTECT:
```

```
ld    (0xFE0AAA), 0xAA      ; 1st Bus Write Cycle
ld    (0xFE0554), 0x55     ; 2nd Bus Write Cycle
ld    (0xFE0AAA), 0xA5     ; 3rd Bus Write Cycle
ld    (XIX), 0x00          ; 4th Bus Write Cycle
```

```
cal   TOGGLECHK             ; トグルビットの確認
cal   PID_ENTRY             ;
ld    A, (XIX)              ; プロテクトアドレスリード
cal   PID_EXIT              ;
cp    A, 0x00               ; (0xFE077E)=0x00?
j     ne, PROTECT_ERR       ; プロテクト状態?
```

PROTECT\_END:

```
j     PROTECT_END           ; プロテクト設定完了
```

PROTECT\_ERR:

```
j     PROTECT_ERR           ; プロテクト設定エラー
```

##### プロダクト ID Entry 処理 #####

PID\_ENTRY:

```
ld    (0xFE0AAA), 0xAA      ; 1st Bus Write Cycle
ld    (0xFE0554), 0x55     ; 2nd Bus Write Cycle
ld    (0xFE0AAA), 0x90     ; 3rd Bus Write Cycle
; --- 300nsec 以上ウェイト(NOP 命令[148nsec/@fPPH=27MHz]を 3 回実行) ---
nop                                     ;
nop                                     ;
nop                                     ; 444nSEC ウェイト
ret                                     ;
```

##### プロダクト ID Exit 処理 #####

PID\_EXIT:

```
ld    (0xFE0000), 0xF0     ; 1st Bus Write Cycle
; --- 300nsec 以上ウェイト(NOP 命令[148nsec/@fPPH=27MHz]を 3 回実行) ---
nop                                     ;
nop                                     ;
nop                                     ; 444nsec ウェイト
ret                                     ;
```

##### トグルビット(D6)確認処理 #####

TOGGLECHK:

```
ld    L, (XIX)             ;
and   L, 0y01000000        ; トグルビット(D6)の確認
ld    H, L                 ; 1 回目のトグルビットデータを保存
```

TOGGLECHK1:

```
ld    L, (XIX)             ;
and   L, 0y01000000        ; トグルビット(D6)の確認
cp    L, H                 ; トグルビット=トグル?
j     z, TOGGLECHK2        ; トグルビットが前回の状態と同じ場合は処理を終了
ld    H, L                 ; 今回の状態を保存
j     TOGGLECHK1           ; 再チェック
```

TOGGLECHK2:

```
ret
```

(プログラム例 : RAM に展開するプログラム)

FE0000H からデータを読み込む

##### フラッシュメモリ読み込み処理 #####

READ:

```
ld    WA, (0xFE0000)       ; フラッシュメモリからデータ読み込み
```

## 4. 電気的特性

### 4.1 絶対最大定格

項目	記号	定格	単位
電源電圧	Vcc	-0.5~4.0	V
入力電圧	VIN	-0.5~Vcc + 0.5	V
出力電流 (1 端子当たり)	IOL (Port8 以外)	2	mA
	IOL (Port8)	20	mA
出力電流 (1 端子当たり)	IOH	-2	mA
出力電流 (合計)	$\Sigma$ IOL (Port8 以外)	60	mA
	$\Sigma$ IOL (Port8)	80	mA
出力電流 (合計)	$\Sigma$ IOH	-80	mA
消費電力 (Ta = 85°C)	PD	600	mW
はんだ付け温度 (10 s)	TSOLDER	260	°C
保存温度	TSTG	-65~150	°C
動作温度	TOPR	-40~85	°C
プログラム消去回数	N <sub>EW</sub>	100	Cycle

注意: 絶対最大定格とは瞬時たりとも超えてはならない規格であり、どの1つの項目も超える事ができない規格です。絶対最大定格を超えると、破壊や劣化の原因となり、破裂・燃焼による傷害を負うことがあります。従って、必ず絶対最大定格を超えないように、応用機器の設計を行ってください。

#### 鉛フリー品 (G 付製品) へのはんだ濡れ性についての注意事項

試験項目	試験条件	備考
はんだ付け性	230°C 5 秒間 1 回 R タイプフラックス使用 (Sn-37Pb 鉛はんだ使用時) 245°C 5 秒間 1 回 R タイプフラックス使用 (Sn-3.0Ag-0.5Cu 鉛フリーはんだ使用時)	フォーミングまでの半田 付着率 95%を良品とする

## 4.2 DC電气的特性 (1/2)

項目	記号	条件		Min	Typ. 注)	Max	単位
電源電圧 ( AVCC = DVCC AVSS = DVSS = 0 V )	VCC	fc = 8~27 MHz	fs = 30~ 34 kHz	2.7		3.6	V
		fc = 8~16 MHz		2.2			
電源電圧 ( AVCC = DVCC AVSS = DVSS = 0 V ) Flash へのプログラムおよび 消去動作時	VCC	fc = 8~27 MHz Ta = -10~40°C		2.7		3.6	V
低レベル 入力電圧	P0, P1, P2, P5, P62, P7, P8, P9, PA, PB	VIL1	Vcc ≥ 2.7 V		-0.3	0.3 Vcc	V
			Vcc < 2.7 V			0.2 Vcc	
	RESET, NMI P60(INT0), P61(INT1)	VIL2	Vcc ≥ 2.7 V			0.25 Vcc	
			Vcc < 2.7 V			0.15 Vcc	
	AM0, AM1	VIL3	Vcc ≥ 2.7 V			0.3	
			Vcc < 2.7 V			0.3	
	X1	VIL4	Vcc ≥ 2.7 V			0.2 Vcc	
			Vcc < 2.7 V			0.1 Vcc	
高レベル 入力電圧	P0,P1,P2,P5,P62,P7,P8, P9,PA,PB	VIH1	Vcc ≥ 2.7 V		0.7 Vcc	Vcc + 0.3	V
			Vcc < 2.7 V		0.8 Vcc		
	RESET, NMI P60(INT0),P61(INT1)	VIH2	Vcc ≥ 2.7 V		0.75 Vcc		
			Vcc < 2.7 V		0.85 Vcc		
	AM0, AM1	VIH3	Vcc ≥ 2.7 V		Vcc - 0.3		
			Vcc < 2.7 V		Vcc - 0.3		
	X1	VIH4	Vcc ≥ 2.7 V		0.8 Vcc		
			Vcc < 2.7 V		0.9 Vcc		
低レベル出力電圧	VOL	IOL = 1.6 mA	Vcc ≥ 2.7 V			0.45	V
		IOL = 0.4 mA	Vcc < 2.7 V			0.15 Vcc	
高レベル出力電圧	VOH	IOH = -400 μA	Vcc ≥ 2.7 V	Vcc - 0.3			V
低レベル出力電流 (Port8)	IOL	VOL = 1.0 V	Vcc ≥ 2.7 V			15	mA
		VOL = 1.0 V	Vcc ≥ 2.2 V			10	

注) Typ.値は特に指定のない限り、Ta = 25°C, Vcc = 3.0 V です。

## DC電气的特性 (2/2)

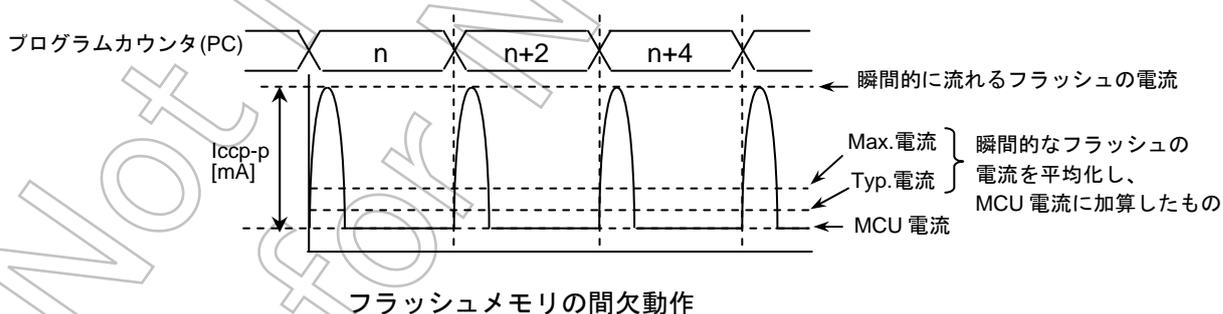
項目	記号	条件	Min	Typ.(注 1)	Max	単位
入力リーク電流	ILI	$0.0 \leq V_{IN} \leq V_{CC}$		0.02	$\pm 5$	$\mu\text{A}$
出力リーク電流	ILO	$0.2 \leq V_{IN} \leq V_{CC} - 0.2$		0.05	$\pm 10$	
パワーダウン電圧 (@STOP, RAMバックアップ)	VSTOP	$V_{IL2} = 0.2 V_{CC}$ , $V_{IH2} = 0.8 V_{CC}$	2.2		3.6	V
RESET プルアップ抵抗	RRST	$V_{CC} = 2.7 \text{ V} \sim 3.6 \text{ V}$	100		400	k $\Omega$
		$V_{CC} = 2.2 \text{ V}$	200		1000	
端子容量	CIO	$f_c = 1 \text{ MHz}$			10	pF
シュミット幅 RESET, NMI, INT0, INT1	VTH	$V_{CC} \geq 2.7 \text{ V}$	0.4			V
		$V_{CC} < 2.7 \text{ V}$	0.3			
NORMAL (注 2)	I <sub>CC</sub>	$V_{CC} = 2.7 \text{ V} \sim 3.6 \text{ V}$ $f_c = 27 \text{ MHz}$		40	50	mA
IDLE2				30	38	
IDLE1				25	30	
NORMAL (注 2)		$V_{CC} = 2.2 \text{ V} \sim 3.6 \text{ V}$ $f_c = 16 \text{ MHz}$		20	28	$\mu\text{A}$
IDLE2				13	18	
IDLE1				9	13	
SLOW (注 2)		$V_{CC} = 2.2 \text{ V} \sim 3.6 \text{ V}$ $f_s = 32.768 \text{ kHz}$		55	75	$\mu\text{A}$
IDLE2				40	60	
IDLE1				35	45	
STOP		$V_{CC} = 2.2 \text{ V} \sim 3.6 \text{ V}$		1	25	$\mu\text{A}$
間欠動作ピーク電流	I <sub>CCP-P</sub>	$V_{CC} = 2.2 \text{ V} \sim 3.6 \text{ V}$		20		mA

注 1) Typ.値は特に指定のない限り、 $T_a = 25^\circ\text{C}$ ,  $V_{CC} = 3.0 \text{ V}$  です。

注 2) I<sub>CC</sub> NORMAL, SLOW の測定条件: すべて動作、出力端子は開放、入力端子はレベル固定。

フラッシュメモリでプログラムが動作しているとき、またはフラッシュメモリからデータをリードしているとき、フラッシュメモリは間欠動作を行いますので、瞬間的に下記タイミング図のようなピーク電流が流れます。よってこの場合の電源電流 I<sub>CC</sub>(NORMAL/SLOW モード時)は、ピーク電流を平均化した電流値と MCU 電流値の和となります。

電源設計の際はピーク電流が供給可能な回路設計にしてください。SLOW モードではピーク電流と平均化された電流の差が大きくなります。



## 4.3 AD 変換特性

AVCC = VCC , AVSS = VSS

項目	記号	条件	Min	Typ.	Max	単位
アナログ基準電圧 (+)	VREFH	$VCC \geq 2.7 V$	$VCC - 0.2V$	VCC	VCC	V
		$VCC < 2.7 V$	VCC	VCC	VCC	
アナログ基準電圧 (-)	VREFL	$VCC \geq 2.7 V$	VSS	VSS	$VSS + 0.2V$	
		$VCC < 2.7 V$	VSS	VSS	VSS	
アナログ入力電圧	VAIN		VREFL		VREFH	
アナログ基準電圧電源電流 <VREFON>=1	IREF (VREFL=0V)	$VCC \geq 2.7 V$		0.94	1.35	
		$VCC < 2.7 V$		0.65	0.90	
<VREFON>=0		$VCC = 2.2 V \sim 3.6 V$		0.02	5.0	$\mu A$
総合誤差 (量子誤差を含まず)	-	$VCC \geq 2.7 V$		$\pm 1.0$	$\pm 4.0$	LSB
		$VCC < 2.7 V$		$\pm 1.0$	$\pm 4.0$	

注1)  $1LSB = (VREFH - VREFL)/1024 [V]$

注2) 最低動作周波数について

AD コンバータの動作は、 $f_c$  (高速発振器) 使用時のみ保証します ( $f_s$  では保証しません)。

注3) AVCC 端子より流れる電源電流は、VCC 端子の電源電流 ( $I_{CC}$ ) に含まれます。

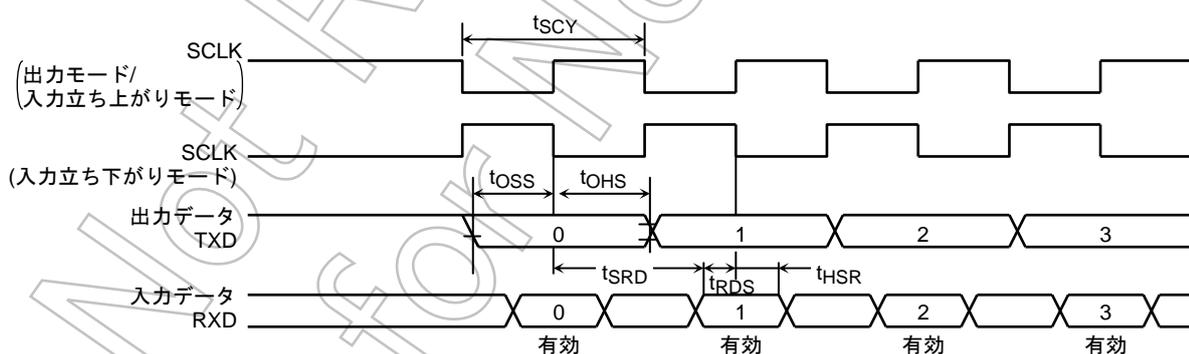
## 4.4 シリアルチャネルタイミング-I/Oインタフェースモード

## (1) SCLK 入力モード

項目	記号	計算式		16 MHz		27 MHz		単位
		Min	Max	Min	Max	Min	Max	
SCLK 周期	t <sub>SCY</sub>	16X		1.0		0.59		μs
出力データ → SCLK 立ち上がり/立ち下がり	t <sub>OSS</sub>	t <sub>SCY</sub> /2 - 4X - 110 (V <sub>CC</sub> = 2.7 V~3.6 V)		140		38		ns
		t <sub>SCY</sub> /2 - 4X - 180 (V <sub>CC</sub> = 2.2 V~2.7 V)		70		-		
SCLK 立ち上がり/立ち下がり → 出力データ 保持	t <sub>OHS</sub>	t <sub>SCY</sub> /2 + 2X + 0		625		370		ns
SCLK 立ち上がり/立ち下がり → 入力データ 保持	t <sub>HSR</sub>	3X + 10		198		121		ns
SCLK 立ち上がり/立ち下がり → 有効データ 入力	t <sub>SRD</sub>		t <sub>SCY</sub> - 0		1000		592	ns
有効データ 入力 → SCLK 立ち上がり/立ち下がり	t <sub>RDS</sub>	0		0		0		ns

## (2) SCLK 出力モード

項目	記号	計算式		16 MHz		27 MHz		単位
		Min	Max	Min	Max	Min	Max	
SCLK 周期	t <sub>SCY</sub>	16X	8192X	1.0	512	0.59	303	μs
出力データ → SCLK 立ち上がり/立ち下がり	t <sub>OSS</sub>	t <sub>SCY</sub> /2 - 40		460		256		ns
SCLK 立ち上がり/立ち下がり → 出力データ 保持	t <sub>OHS</sub>	t <sub>SCY</sub> /2 - 40		460		256		ns
SCLK 立ち上がり/立ち下がり → 入力データ 保持	t <sub>HSR</sub>	0		0		0		ns
SCLK 立ち上がり/立ち下がり → 有効データ 入力	t <sub>SRD</sub>		t <sub>SCY</sub> - 1X - 180		757		375	ns
有効データ 入力 → SCLK 立ち上がり/立ち下がり	t <sub>RDS</sub>	1X + 180		243		217		ns



注 1) SCLK 立ち上がり/立ち下がり…SCLK 立ち上がりモードの場合は SCLK 立ち上がり、SCLK 立ち下がりモードの場合は SCLK 立ち下がりのタイミングです。

注 2) 27 MHz, 16 MHz の値は、t<sub>SCY</sub> = 16X のときの値です。

注 3) 表中の「x」は、クロック f<sub>FPH</sub> の周期を示します。f<sub>FPH</sub> の周期は、CPU コアで使用されるシステムクロック f<sub>SYS</sub> 周期の 1/2 です。

クロック f<sub>FPH</sub> の周期は、高速発振器/低速発振器の切り替えなどに依存します。

## 4.5 タイマカウンタ入力(ECIN)特性

項目	記号	条件		Min	Typ.	Max	単位
タイマカウンタ入力 (ECIN1~ ECIN3 入力)	$t_{TC1}$	周波数測定モード VDD = 2.7 ~ 3.6V	片エッジカウント	-	-	$f_c/2$ ただし $f_c/2=Max8MHz$	MHz
			両エッジカウント				
		周波数測定モード VDD = 2.2 ~ 2.7V	片エッジカウント	-	-		
			両エッジカウント				

## 4.6 割り込み

(1)  $\overline{NMI}$ , INT0, INT1 割り込み

項目	記号	計算式		16 MHz		27 MHz		単位
		Min	Max	Min	Max	Min	Max	
$\overline{NMI}$ , INT0, INT1 低レベルパルス幅	$t_{INTAL}$	$4X + 40$		290		188		ns
$\overline{NMI}$ , INT0, INT1 高レベルパルス幅	$t_{INTAH}$	$4X + 40$		290		188		ns

注 1)  $X_c$  は、高速発振器側のクロック  $f_c$  の周期を示します。

注 2) 表中の「X」は、クロック  $f_{FPH}$  の周期を示します。 $f_{FPH}$  の周期は、CPU コアで使用されるシステムクロック  $f_{SYS}$  周期の 1/2 です。

クロック  $f_{FPH}$  の周期は、高速発振器/低速発振器の切り替えなどに依存します。

## 4.7 フラッシュ特性

(1) 書き込み特性

項目	条件	Min	Typ.	Max	単位
フラッシュメモリ書き替え保証回数	VCC = 2.7 ~ 3.6V $f_c = 8MHz \sim 27MHz$ $T_a = -10 \sim 40^\circ C$	-	-	100	回

## 4.8 推奨発振回路

TMP91FW40FG は、下記の発振子メーカーにて評価されております。発振子の選択時に活用願います。

注) 発振端子のトータル負荷容量は、接続する外付け（または内蔵）負荷容量 C1, C2 と実装基板上の浮遊容量の和になります。C1, C2 の定数を使用した場合でも、実装基板により負荷容量が異なり発振器が誤動作する可能性があります。基板設計の際は発振回路周辺のパターンが最短距離になるようにしてください。また、実際に使用される実装基板での発振評価を行うことを推奨いたします。

### (1) 接続例

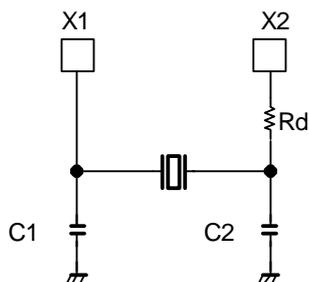


図 4.8.1 高周波発振器の接続図

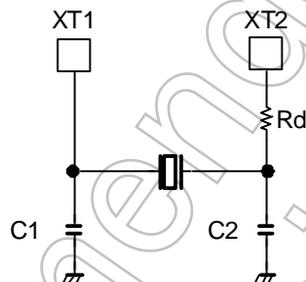


図 4.8.2 低周波発振器の接続図

### (2) TMP91FW40FG 推奨セラミック発振子

本製品は(株)村田製作所社製セラミック発振子を推奨しております。  
詳細につきましては、下記 URL の同社ホームページを参照してください。  
<http://www.murata.co.jp>

## 5. ポート部等価回路図

- 回路図の見方

基本的に、標準 CMOS ロジック IC 「74HCxx」 シリーズと同じゲート記号を使って書かれています。

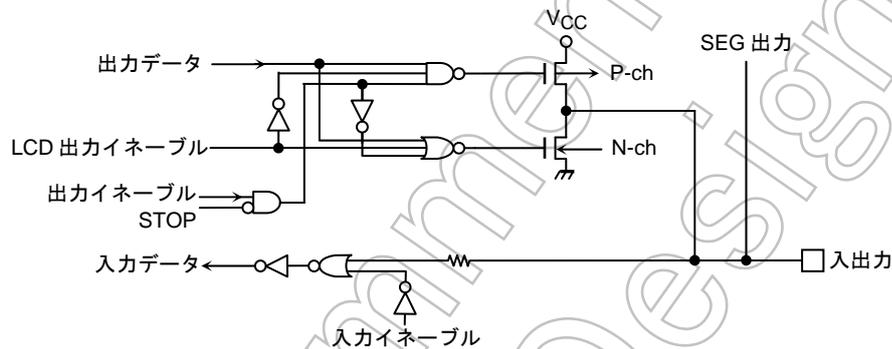
信号名の中で、特殊なものについては、下記に示します。

**STOP**: この信号は、HALT モード設定レジスタを「STOP」モード (SYSCR2<HALTM1:0>="01") にして、CPU が「HALT」命令を実行したとき、アクティブ“1”になります。

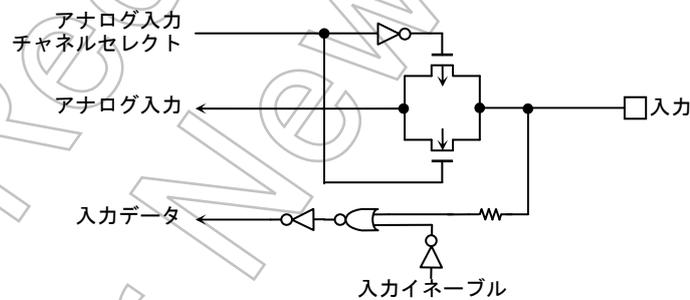
ただし、ドライバイネーブルビット SYSCR2<DRVE> が“1”にセットされているときは、STOP は“0”のままです。

- 入力保護抵抗は、数十Ω ~ 数百Ω 程度です。

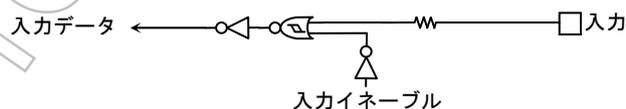
- P0 (SEG24~SEG31), P1 (SEG16~SEG23), P2 (SEG8~SEG15), PB (SEG32~SEG39)



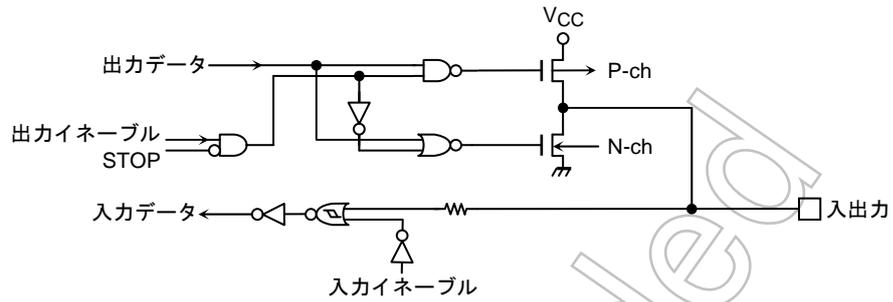
- P5 (AN0~AN3/KWI0~KWI3)



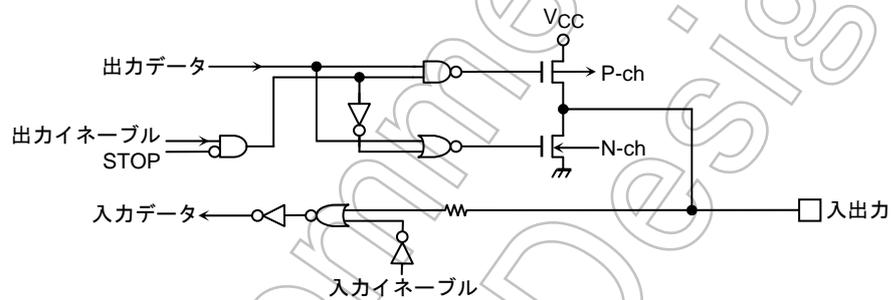
- P60 (INT0)



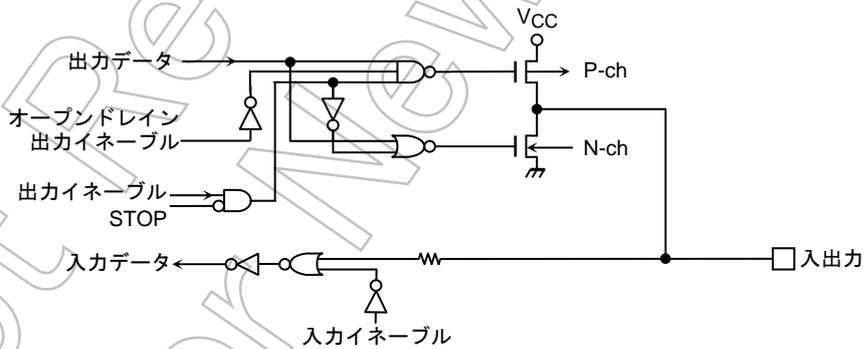
■ P61 (INT1)



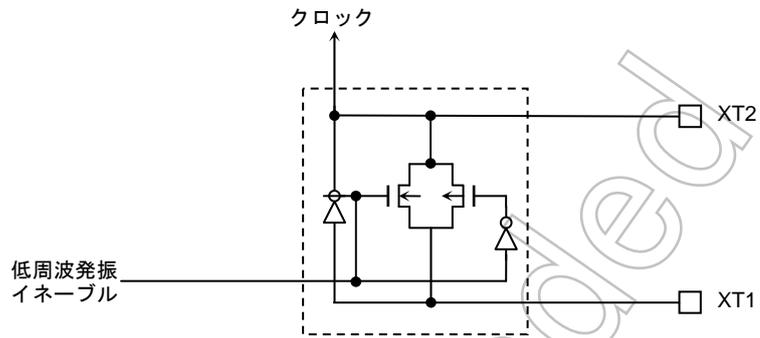
■ P62(ALARM), P70~P75(ECNT1~ECNT3, ECIN1~ECIN3), P91(RXD0), P92(SCLK0/CTS0), P94 (RXD1), P95(SCLK1/CTS1), PA1(RXD2), PA2(SCLK2/CTS2), PA4(RXD3), PA5(SCLK3/CTS3)



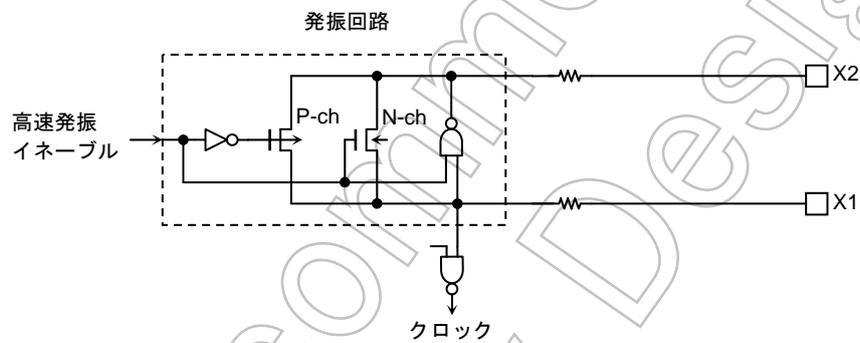
■ P80~P83(TC5OUT~TC8OUT), P90(TXD0), P93(TXD1), PA0(TXD2), PA3(TXD3)



■ XT1, XT2



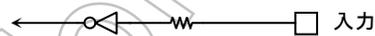
■ X1, X2



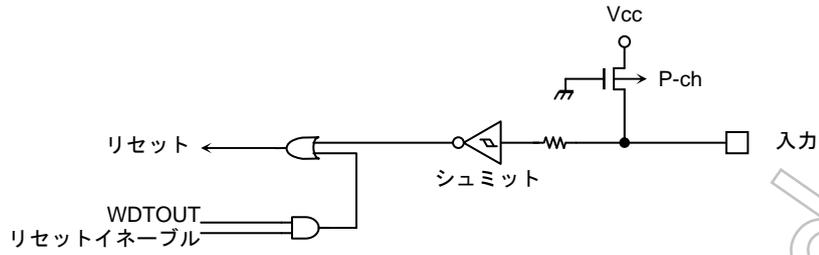
■  $\overline{\text{NMI}}$



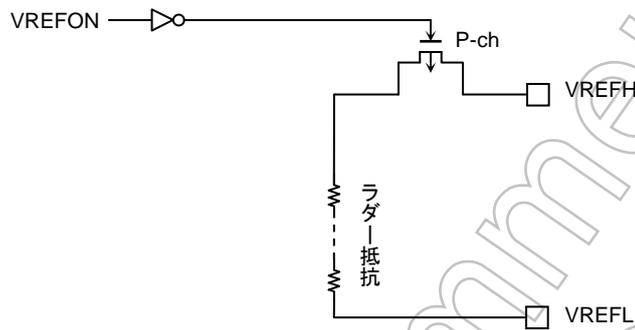
■ AM0~AM1



■  $\overline{\text{RESET}}$



■ VREFH, VREFL



Not Recommended for New Design

6. パッケージ外形寸法図

LQFP100-P-1414-0.50F

単位: mm

