

シリアル I²C nvSRAM を使った設計

作成者: Shivendra Singh

関連プロジェクト: あり

関連製品ファミリ: CY14xxxxI、CY14xxxxJ

ソフトウェア バージョン: PSoC[®] Creator™ 3.0 以降

関連アプリケーション ノート: [AN61546](#)、[AN43593](#)

AN74875 は、I²C nvSRAM デバイスの設計ガイドラインと回路例を提供します。I²C nvSRAM は、ゼロサイクル遅延の書き込み動作と無制限の SRAM 書き換え回数を備えた高性能の不揮発性シリアル メモリです。I²C nvSRAM はスレーブ I²C デバイスであり、システムにアクセスするために I²C のマスター コントローラが必要です。PSoC3 の関連ライブラリ コンポーネントもサンプル プロジェクトとして提供されます。

目次

はじめに.....	1
I ² C nvSRAM のコンフィギュレーション	2
I ² C バス プロトコル機能の適用性	2
I ² C nvSRAM デバイスのオプション	2
I ² C nvSRAM デバイスの接続.....	3
I ² C のプルアップ抵抗値を決定	5
制御入力ピンのコンフィギュレーション	7
RTC 部固有のピン配置.....	8
I ² C nvSRAM の動作.....	9
高速モード (Hs モード) の動作.....	10
I ² C nvSRAM でのアドレス指定.....	10
I ² C nvSRAM のアクセス	12
まとめ.....	16
付録 A (疑似コード例).....	17
I ² C の書き込み	17
I ² C の読み出し.....	18
ワールドワイドな販売と設計サポート	20

はじめに

サイプレスの nvSRAM は、SRAM セルと不揮発性メモリ セルを単一の nvSRAM セルに統合しています。通常の動作モードでは、全ての読み出しと書き込みは nvSRAM の SRAM 部からの読み出しと SRAM 部への書き込みが直接行われます。これは EEPROM やフラッシュなど既存の不揮発性メモリ技術に比べて、より高速な読み書きアクセスを提供します。システムの電源喪失が発生した場合、SRAM からのデータは、V_{CAP} ピンに接続されている小さなコンデンサに蓄積されたエネルギーを使用して、不揮発性セルに自動的に転送されます。次の電源投入サイクル中に、不揮発性セルからのデータは自動的に SRAM アレイにリコールされ、利用できるようになります。V_{CAP} ピンに接続されたコンデンサは通常処理中に nvSRAM によって充電されます。

nvSRAM は、その不揮発性セルに対して 100 万回の書き換え回数を仕様として定めています。nvSRAM での書き換え回数は、STORE 処理を行っている間にデータが SRAM セルから不揮発性セルに転送された場合にのみカウントされます。nvSRAM の不揮発性ストア処理は、デバイスの電源が所定の閾値レベル (V_{SWITCH}) を下回った時に自動的に開始するか、またはコマンドレジスタ (0xAA) にコマンドを書き込むか、あるいはハードウェアピン (HSB) を LOW にトグルすることで要求に応じて開始します。nvSRAM のコマンド レジスタは制御レジスタ空間で定義されています。制御レジスタは専用の I²C スレーブ ID を介してアドレス指定されます。nvSRAM のアドレス指定の詳細は、本アプリケーション ノートの後半に説明する I²C nvSRAM の節に記載されています。

システムの電源障害が検出され、SRAM に書き込まれた新しいデータが不揮発性セルに安全に移動する必要がある場合にのみ、nvSRAM は不揮発性ストアを開始します。したがって、nvSRAM で測定される不揮発性セルの合計書き換え回数は、SRAM 書き込みサイクルではなく、不揮発性ストア サイクルの合計になります。

電力喪失が発生した場合に実行時の重要な情報を瞬時に保存することを必要とする多くのデータ ロギング アプリケーションがあり

ます。この重要な情報には、コントローラの実行時の状態、スクラッチ パッド データ、パラメータ設定およびコントローラによって測定された他の環境変数が含まれています。I²C nvSRAM はその高速不揮発性書き込み速度によって、そのようなデータ ロギングアプリケーションに理想的に適合することができます。nvSRAM では、I²C マスター コントローラが数十マイクロ秒で数百バイトのデータをログできるのに対して、EEPROM やフラッシュ メモリでは同じ量のデータを書き込むには数十ミリ秒が必要です。I²C nvSRAM は、業界標準の 8 ピン SOIC と 16 ピン SOIC パッケージで提供されています。

このアプリケーション ノートでは、I²C nvSRAM のコンフィギュレーション、異なるパッケージ オプションの回路例、I²C バスのプルアップ抵抗の適切な値を決定する方法、nvSRAM での I²C 通信のデータ バイト形式、メモリにアクセスするための I²C アドレス指定機構、リアル タイム クロック(RTC)、および nvSRAM の制御機能について説明します。I²C nvSRAM に関する他の詳細については、固有のデバイス データシートを参照してください。

PSoC 3 nvRAM I²C のライブラリ コンポーネントは関連プロジェクトとして本アプリケーション ノートに添付されています。

I²C nvSRAM のコンフィギュレーション

I²C nvSRAM は、最高 3.4M ビット/秒のレート (I²C クロック周波数が 3.4MHz) で I²C のデータ転送をサポートし、I²C バス規格の仕様で定義された他の全ての低周波数アクセスもサポートしています。

- **標準モード (Sm)** - ビット レートが最大 100Kbit/s
- **ファースト モード (Fm)** - ビット レートが最大 400Kbit/s
- **ファースト モード プラス (Fm+)** - ビット レートが最大 1Mbit/s

表 2. I²C nvSRAM のコンフィギュレーション

nvSRAM の製品番号	状況	動作電圧 (標準値)	パッケージ	WP ピン	V _{CAP} ピン / AutoStore	(HSB) ピン / HW ストア	A0 ピン	I ² C バスに対するデバイス数	RTC
CY14CXXXJ1	NRND	2.5V	8 SOIC	有	無 / 無	無 / 無	有 ^{注1}	4 または 8 ^{注1}	無
CY14BXXXJ1	NRND	3V							
CY14EXXXJ1	NRND	5V							
CY14CXXXJ2	NRND	2.5V	8 ピン SOIC	有	有 / 有	無 / 無	無	4	無
CY14BXXXJ2	NRND	3V							
CY14EXXXJ2	NRND	5V							
CY14CXXXJ3	NRND	2.5V	16 ピン SOIC	有	有 / 有	有	有 ^{注1}	4 または 8 ^{注1}	無
CY14BXXXJ3	NRND	3V							
CY14EXXXJ3	NRND	5V							
CY14CXXXI	サイプレスにお問い合わせ	2.5V	16 ピン SOIC	有	有 / 有	有	有 ^{注1}	4 または 8 ^{注1}	有
CY14BXXXI	製造中	3V							
CY14EXXXI	製造中	5V							

NRND – 新規設計用にはお勧めできません。

注 1: 最下位のスレーブ アドレス ビット空間 (A0) は 1M ビットの nvSRAM デバイスで内部で使用されているため、1M ビットのメモリ容量オプションでは使用できません。A0 ピンは J2 部を除き、全ての 512K ビット以下のメモリ容量オプションで利用可能です。A0 ピンがないと、I²C nvSRAM は、I²C バスに対して最大 4 個のデバイスに制限されます。

■ 高速モード (Hs) - ビット レートが最大 3.4Mbit/s

上記の 4 バス モードは全てのデバイス コンフィギュレーションで提供されており (表 1 を参照)、デバイスでの特別な設定を必要としません。

I²C バス プロトコル機能の適用性

表 1 は、標準的な I²C のスレーブ バス仕様のすべての必須の機能と任意選択の機能をまとめたものです。I²C nvSRAM は、標準的な I²C のスレーブ デバイスのすべての必須機能に対応しています。

表 1. I²C バス プロトコルの適用性

機能	I ² C 仕様規格	I ² C nvSRAM
START 状態	必須	√
STOP 状態	必須	√
アクノリッジ	必須	√
7 ビット スレーブ アドレ	必須	√
10 ビット スレーブ アドレ	任意に選択	未提供
クロック ストレッチ	任意に選択	不要
一般的な呼び出しアドレ	任意に選択	未提供
デバイス ID	任意に選択	未提供
ソフトウェア リセット	任意に選択	未提供

I²C nvSRAM デバイスのオプション

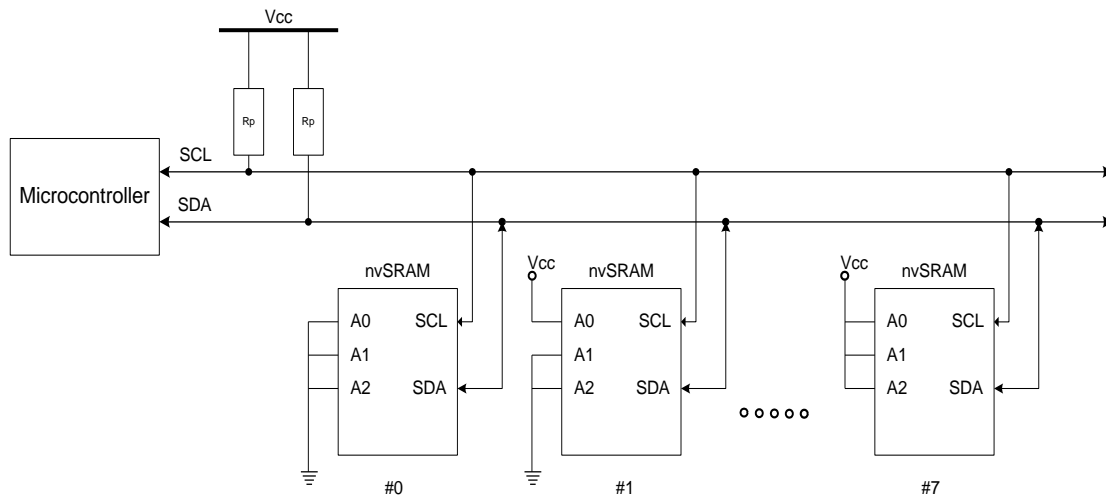
表 2 に示すように、サイプレスは異なるコンフィギュレーションとパッケージ オプションで I²C nvSRAM に対応します。

I²C nvSRAM デバイスの接続

標準的な I²C のシングル マスター マルチ スレーブのコンフィギュレーションは図 1 に示します。I²C のマスター デバイスは、I²C のマスター プロトコルを生成できるマイクロコントローラまたはプログラム可能なデバイスのいずれかであり、スレーブ デバイスは標準的な I²C のスレーブ デバイスのいずれかです。図 1 の例では、I²C nvSRAM は I²C のスレーブ デバイスとして扱われます。512K ビット以下のメモリ容量の I²C nvSRAM は、いくつかのパッケージ オプションで 3 つのスレーブ アドレス指定ビット

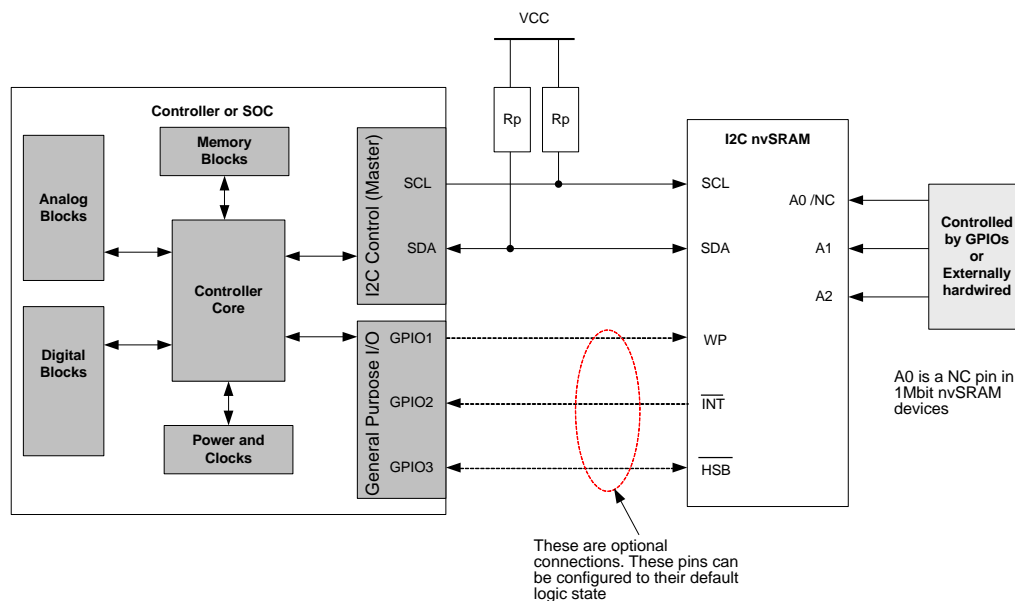
をサポートしているため、同じ I²C バスで最大 8 個のデバイスを接続できます。ユニークなスレーブ ID は、スレーブ 選択 アドレス線 (A2、A1、A0) を 8 種類の異なる組み合わせで設定することにより各スレーブ デバイスに割り当てられます。A0 が使用できないパッケージ コンフィギュレーションでは、スレーブ 選択アドレス ピン A2 と A1 を設定することで同じバスを共有する最大 4 個のスレーブ デバイスのみ接続できます。

図 1. 標準的な I²C のマスター スレーブのコンフィギュレーション



I²C nvSRAM デバイスの一般的なシステム レベルのコンフィギュレーションは図 2 に示します。専用 I²C バスを持たないマイクロコントローラの場合、汎用 I/O ポートはビット バンギングによって SCL と SDA 用に使用されます。

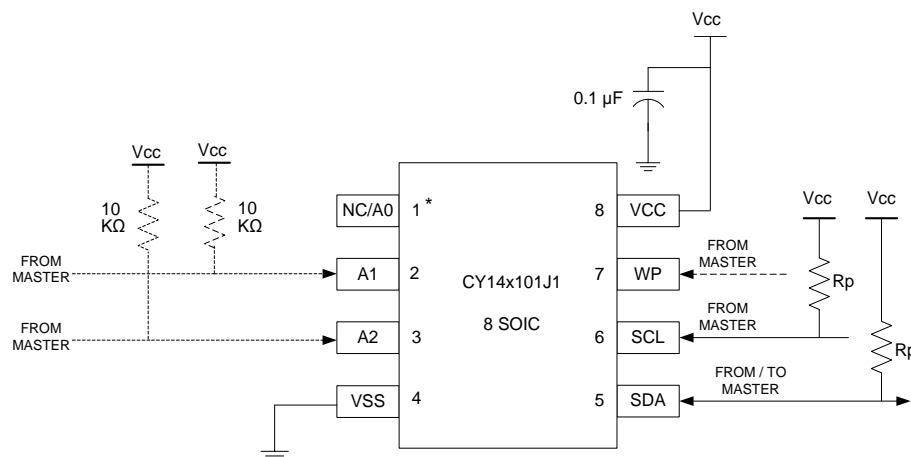
図 2. 一般的な I²C nvSRAM の接続



回路例

以下の図 (図 3~5) は、1Mビット I²C nvSRAM の詳しい回路図の接続を示します。I²C マスターと nvSRAM スレーブ間のハードウェア接続は全ての小メモリ容量 (512Kビット以下) 部品で同じです。

図 3. 8ピン SOIC の 1Mビット I²C nvSRAM のインターフェース (V_{CAP} なし)



全ての任意選択の接続は点線で示しています。

A2 と A1 上のプルアップ抵抗は、スレーブ アドレスビット A2 と A1 を「1」に設定します。システムがアドレス ピンのいずれかを「0」に設定するよう要求する場合は、プルアップ抵抗を取り外して、ピンをフローティング状態にします。弱プルダウン抵抗によって内部で LOW にプルされます。

図 4. 8ピン SOIC の 1Mビット I²C nvSRAM のインターフェース (V_{CAP} 付き)

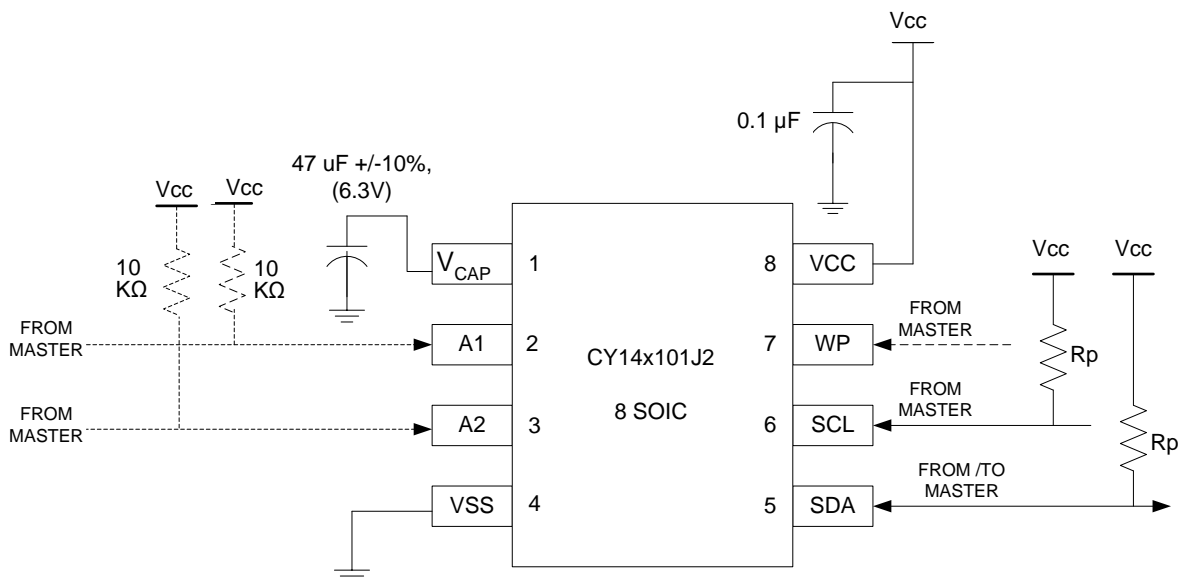
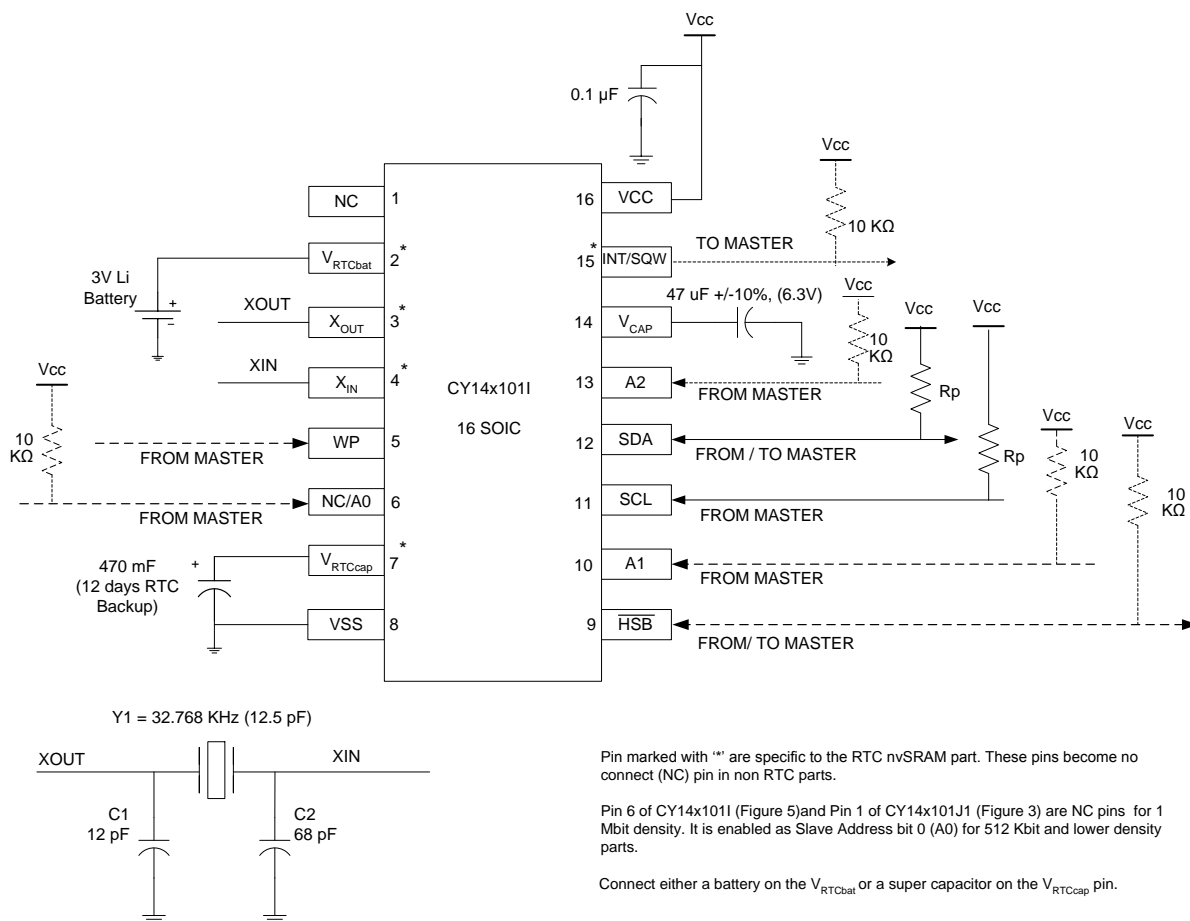


図 5. 16 ピン SOIC の 1Mビット (RTC) I²C nvSRAM のインターフェース



I²C のプルアップ抵抗値を決定

I²C バスは SDA と SCL ライン上にデータとクロックを伝送します。SDA と SCL ラインはオープンドレイン (TTL ファミリではオープンコレクタとしても知られている) 出力ドライバであり、即ち、I²C マスターとスレーブ デバイスはこれらのラインを論理 LOW にのみ駆動するか、または開放したままにできます。同じバスにある I²C デバイスがラインを LOW にプルしていない場合、終端抵抗 (R_p) はラインの HIGH レベルを得るのに V_{CC} にプルします。マルチ マスターのコンフィギュレーションやスレーブによるクロック ストレッチなどの特別な I²C の機能に対応するためにオープンドレイン ドライバのコンフィギュレーションが必要です。クロック ストレッチは I²C 規格の任意選択機能であり、I²C nvSRAM ではサポートされていないため、I²C のクロック信号は I²C nvSRAM では入力 (のみ) 信号になります。

合計バス容量 (Cb) と共に終端抵抗 (Rp) は SDA と SCL の信号のタイミング動作に影響を与えます。I²C デバイスはオープンドレインのドライバでラインをプルダウンしている間に、プ

プルアップ抵抗 R_p は指定された時間内に信号を HIGH レベルに戻す役割があります。プルアップ抵抗 R_p の値は、動作電圧 (V_{CC})、デバイスの出力 LOW 論理レベル (V_{OL}) 仕様、即ち吸い込み電流 (I_{OL}) 仕様、バスの合計通信容量 (C_b) および立ち上がり時間 (t_R) 仕様のようなタイミング パラメータなど、複数の電気的パラメータに依存します。

次の節では、特定のシステム コンフィギュレーションで I²C バスのプル抵抗の値を決定する方法について説明します。

Rp (Max) の決定

RC 時定数の計算のために、CMOS 論理レベルの入力閾値を $V_{IH} = 0.7V_{CC}$ (Min) および $V_{IL} = 0.3V_{CC}$ (Max) と考えてみます。

$V(t) = V_{CC} (1 - e^{-t/RC})$ 、ここで「 t 」は充電開始時からの時間であり、 RC は時定数です。

$V(t_1) = 0.3 \times V_{CC} = V_{CC} (1 - e^{-t_1 / RC})$ のとき:

$$t1 = 0.3566749 \times RC \quad \text{式 1}$$

$V(t_2) = 0.7 \times V_{CC} = V_{CC} (1 - e^{-t_2 / RC})$ のとき:

$$t_2 = 1.2039729 \times RC \quad \text{式 2}$$

合計の立ち上がり時間 (T) は、バス容量の電圧レベルを V_{IL} から V_{IH} に充電するのに要する時間です:

$$\begin{aligned} T &= t_2 - t_1 \\ &= 1.2039729 \times RC - 0.3566749 \times RC \\ &= 0.8473 \times RC \quad \text{式 3} \end{aligned}$$

式 3 は、I²C ラインに接続するためにプルアップ抵抗値の上限を決定するために使用されます。表 3 は、全てのタイミングモードに対してバス容量の関数として最大 R_p を示します。各モードに対して $R_p(\text{max})$ は、最小立ち上がり時間 (t_R) と推定されるバス容量 (C_b) の関数です:

$$R_p(\text{Max}) = \frac{t_R}{(0.8473 \times C_b)} \quad \text{式 4}$$

バス容量 (C_b) は、配線部、接続部、端子部の合計容量です。

$R_p(\text{Min})$ の決定

動作電圧と吸い込み電流 (I_{OL}) はプルアップ抵抗の最小値、 $R_p(\text{min})$ を制限します。 V_{CC} と I_{OL} の関数である $R_p(\text{Min})$ は式 5 を使って計算されます。

$$R_p(\text{min}) = \frac{V_{CC} - V_{OL}(\text{max})}{I_{OL}} \quad \text{式 5}$$

R_p の値は、仕様の最小値と最大値の範囲内で選択する必要があります。

$$R_p(\text{Min}) \leq R_p \leq R_p(\text{Max}) \quad \text{式 6}$$

低消費電力の設計では、電流消費量を制限するために範囲の上限に向かう値を使って選ぶ必要があります。

表 3 は、特定のバス負荷条件と動作電圧に対応する R_p の値 (Min, Max) の一覧を提供しています。表 3 に表示していない値は、 $R_p(\text{Max})$ と $R_p(\text{Min})$ を計算するための式 4 と式 5 から得られます。

表 3 の網掛け領域は、特定の動作電圧の条件下でいくつかのバス負荷 (C_b) に対して $R_p(\text{Min})$ が $R_p(\text{Max})$ を超えていることを示します。 $R_p(\text{Min})$ が $R_p(\text{Max})$ を超えることができないため、これは I²C バスに使用される最大の容量負荷 (C_b) を制限します。

例えば: 3V の部品が最小 V_{CC} 電源 ($V_{CC} = 2.7V$) で動作するように設定された場合、システムが以下のバス モードで動作する時、SCL と SDA ライン上に以下に示す負荷 (pF 単位) を超えてはいけません:

$$S_m = C_b \leq 550 \text{ pF}; 0.77 \text{ k}\Omega \leq R_p \leq 2.15 \text{ k}\Omega$$

$$F_m = C_b \leq 450 \text{ pF}; 0.77 \text{ k}\Omega \leq R_p \leq 0.79 \text{ k}\Omega$$

$$F_{m+} = C_b \leq 150 \text{ pF}; 0.77 \text{ k}\Omega \leq R_p \leq 0.94 \text{ k}\Omega$$

$$H_s = C_b \leq 100 \text{ pF}; 0.77 \text{ k}\Omega \leq R_p \leq 0.94 \text{ k}\Omega$$

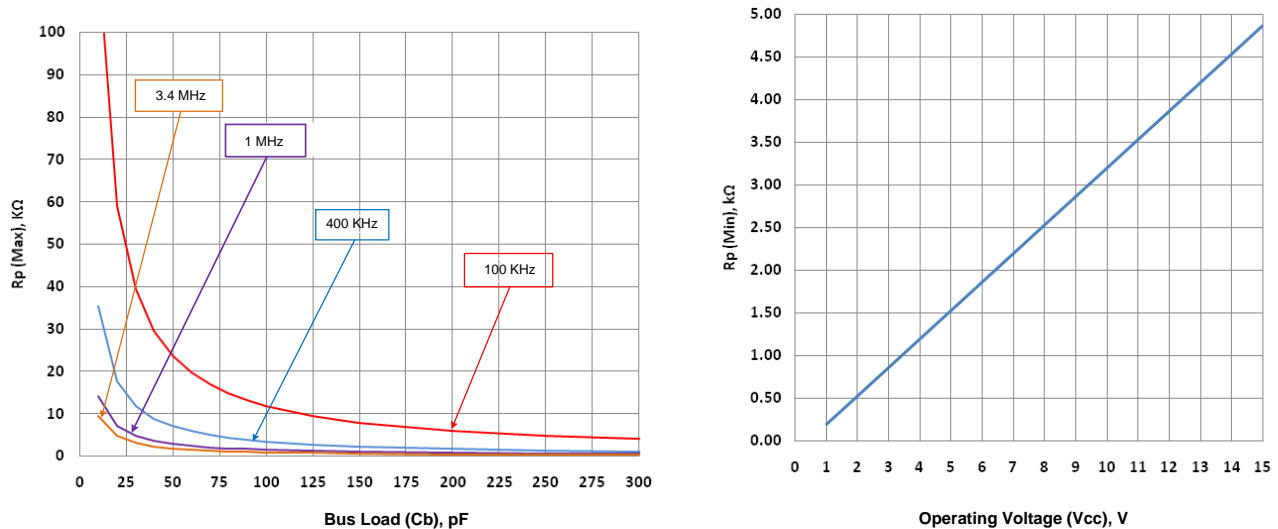
同様に、他の動作電圧と動作周波数に対応した最大のバス負荷 (C_b) と I²C のプルアップ抵抗 (R_p) の値が表 3 と図 6 から得られます。

表 3. 異なるバス負荷と動作電圧に対応する $R_p(\text{Min}, \text{Max})$

C_b (pF)	$R_p(\text{Min})$ (k Ω)	$R_p(\text{Max})$ (k Ω)			
		100 KHz	400 KHz	1 MHz	3.4 MHz
10	0.68	118.02	35.41	14.16	9.44
20	0.68	59.01	17.70	7.08	4.72
30	0.68	39.34	11.80	4.72	3.15
40	0.68	29.51	8.85	3.54	2.36
50	0.68	23.60	7.08	2.83	1.89
60	0.68	19.67	5.90	2.36	1.57
70	0.68	16.86	5.06	2.02	1.35
80	0.68	14.75	4.43	1.77	1.18
90	0.68	13.11	3.93	1.57	1.05
100	0.68	11.80	3.54	1.42	0.94
125	0.68	9.44	2.83	1.13	0.76
150	0.68	7.87	2.36	0.94	0.63
200	0.68	5.90	1.77	0.71	0.47
250	0.68	4.72	1.42	0.57	0.38
300	0.68	3.93	1.18	0.47	0.31
350	0.68	3.37	1.01	0.40	0.27
400	0.68	2.95	0.89	0.35	0.24
450	0.68	2.62	0.79	0.31	0.21
500	0.68	2.36	0.71	0.28	0.19
550	0.68	2.15	0.64	0.26	0.17

C_b (pF)	$R_p(\text{Min})$ (k Ω)	$R_p(\text{Max})$ (k Ω)			
		100 KHz	400 KHz	1 MHz	3.4 MHz
10	0.77	118.02	35.41	14.16	9.44
20	0.77	59.01	17.70	7.08	4.72
30	0.77	39.34	11.80	4.72	3.15
40	0.77	29.51	8.85	3.54	2.36
50	0.77	23.60	7.08	2.83	1.89
60	0.77	19.67	5.90	2.36	1.57
70	0.77	16.86	5.06	2.02	1.35
80	0.77	14.75	4.43	1.77	1.18
90	0.77	13.11	3.93	1.57	1.05
100	0.77	11.80	3.54	1.42	0.94
125	0.77	9.44	2.83	1.13	0.76
150	0.77	7.87	2.36	0.94	0.63
200	0.77	5.90	1.77	0.71	0.47
250	0.77	4.72	1.42	0.57	0.38
300	0.77	3.93	1.18	0.47	0.31
350	0.77	3.37	1.01	0.40	0.27
400	0.77	2.95	0.89	0.35	0.24
450	0.77	2.62	0.79	0.31	0.21
500	0.77	2.36	0.71	0.28	0.19
550	0.77	2.15	0.64	0.26	0.17

C_b (pF)	$R_p(\text{Min})$ (k Ω)	$R_p(\text{Max})$ (k Ω)			
		100 KHz	400 KHz	1 MHz	3.4 MHz
10	1.37	118.02	35.41	14.16	9.44
20	1.37	59.01	17.70	7.08	4.72
30	1.37	39.34	11.80	4.72	3.15
40	1.37	29.51	8.85	3.54	2.36
50	1.37	23.60	7.08	2.83	1.89
60	1.37	19.67	5.90	2.36	1.57
70	1.37	16.86	5.06	2.02	1.35
80	1.37	14.75	4.43	1.77	1.18
90	1.37	13.11	3.93	1.57	1.05
100	1.37	11.80	3.54	1.42	0.94
125	1.37	9.44	2.83	1.13	0.76
150	1.37	7.87	2.36	0.94	0.63
200	1.37	5.90	1.77	0.71	0.47
250	1.37	4.72	1.42	0.57	0.38
300	1.37	3.93	1.18	0.47	0.31
350	1.37	3.37	1.01	0.40	0.27
400	1.37	2.95	0.89	0.35	0.24
450	1.37	2.62	0.79	0.31	0.21
500	1.37	2.36	0.71	0.28	0.19
550	1.37	2.15	0.64	0.26	0.17

図 6. 異なるバス負荷と動作電圧に対応する R_p (Min, Max)


制御入力ピンのコンフィギュレーション

I²C nvSRAM は入力ピンである数多くの制御ピンを備えていて、それらのピンはデバイスの正常な動作のために固定された論理状態 (HIGH または LOW) に適切にバイアスされる必要があります。制御入力ピンが適切な論理レベル (HIGH または LOW のいずれか) にバイアスされずにフローティング状態になっていると、フローティング ピンがデバイス動作をランダムにするいくつかの中間準安定状態に定まることがあります。そのため、内部プルアップまたはプルダウン オプションがない未使用の入力ピンは、常にプルアップまたはプルダウン抵抗を使用して外部的に適切な論理レベルに接続する必要があります。1 kΩ~10 kΩ の抵抗がこの目的に使用されます。

WP ピン:

WP ピンはアクティブ HIGH ピンであり、書き込み処理からメモリ全体と全てのレジスタを保護します。このピンが HIGH になると、全てのメモリとレジスタ書き込みは禁止され、アドレス カウンタはインクリメントされません。I²C nvSRAM では、このピンに内部プルダウン抵抗を取り付けています。したがって、書き込み保護機能が使われない場合、このピンはフローティング状態 (未接続) にできます。このピンが外部制御用にコントローラ I/O に接続される場合は、外部プルアップ抵抗が推奨されて、このライン上のノイズに起因する望ましくないトリガを防ぎます。1 kΩ~10 kΩ の抵抗がこの目的に使用されます。

A2、A1、A0 ピン:

これらはスレーブ アドレス ピンであり、マルチ スレーブのコンフィギュレーションに際し異なるスレーブ デバイスに対応した異なるスレーブ アドレスを設定するために使用されます。これらのピンは内部的に LOW にプルされるため、使用されない場合はフローティング (未接続) のままにできます。論理 HIGH 状態に設定するために、これらのピンは外部プルアップ抵抗に接

続するか、または直接 V_{CC} 電源に接続する必要があります。1kΩ~10kΩ の抵抗がプルアップ抵抗として使用できます。システムがスレーブ アドレスをダイナミックに変更するよう要求する幾つかのコンフィギュレーションでは、これらのアドレス ピンは、実行中にスレーブ選択アドレス ピン (A2、A1、A0) を設定するためにコントローラ I/O に接続し、デバイスにアクセスする必要があります。

HSB ピン:

HSBピンは nvSRAM の双方向ピンです。出力ピンとして動作する場合、このピンは通常処理中は nvSRAM のレディまたはビジー状態にします。デバイスの電源投入時または不揮発性ストア サイクルが進行中には、HSBピンはそのビジー状態を表す示すデバイスによって LOW にプルされます。HSBピンは HIGH 状態になると、デバイスが通常の読み書き処理に応じたレディ状態にあることを示します。入力ピンとして動作する場合、HSBピンはコントローラで LOW にプルすることで外部でハードウェア STORE を開始するのに使用されます。このピンは任意の GPIO に接続されていない場合、フローティングのままにすることができます。I²C nvSRAM は、通常動作中に HSB ピンを HIGH に保つために、このピンに内部の弱プルアップ抵抗を用意しています。このピンが外部制御用にコントローラ I/O に接続される場合は、外部プルアップ抵抗が推奨されて、このライン上のノイズに起因する望ましくないトリガを防ぎます。1 kΩ~10 kΩ の抵抗がこの目的に使用されます。

V_{CAP}:

V_{CAP} ピンに接続されたコンデンサは、電力喪失時にデータを SRAM から不揮発性素子に転送するために nvSRAM に電源を供給します。通常処理中に、デバイスは V_{CC} から電流を引き込んで、V_{CAP} ピンに接続されたコンデンサを充電します。V_{CAP} での蓄積された電荷は、nvSRAM デバイスが一回の STORE

処理を実行するために使用されます。V_{CC} ピンの電圧が V_{SWITCH} を下回ると、デバイスは自動的に V_{CAP} ピンを V_{CC} から切り離し、STORE 処理は V_{CAP} の蓄積電荷を使用して開始されます。

正常な AutoStore 処理のために、V_{CAP} ピンに適切な値のコンデンサを接続する必要があります。選択したコンデンサの値は、デバイスのデータシートに規定された範囲内である必要があります。コンデンサの不適切な選択はデバイスの誤動作につながる可能性があります。nvSRAM 製品用のコンデンサ選択のガイドラインの詳細については、アプリケーション ノート [AN43593 - Storage Capacitor Options for Cypress nvSRAM](#) を参照してください。

RTC 部固有のピン配置

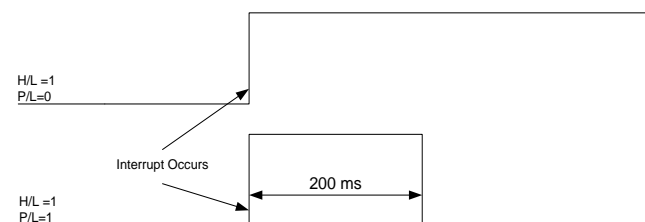
RTC 機能はパッケージに以下の追加ピンを必要とします。正常な RTC 機能のために、これらのピンは適切に設定される必要があります。

INT ピン: これは RTC 部の出力ピンです。RTC nvSRAM は、アラーム、ウォッチドッグタイマ、校正クロック出力、および方形波ジェネレータなど様々な機能を提供しています。INT 出力は、nvSRAM 内で定義された RTC レジスタの設定とそれらの優先順位に応じて、機能面の状態／出力を引き出すために多重化されています。INT ピンは設定可能なドライバ出力です。INT ピンの出力モードは、I²C スレーブ デバイス内の割り込み状態／制御レジスタ (0x06) の「H/L」ビットをセットすることで設定されます。H/L ビットを「1」にセットすると、INT 出力はアクティブ HIGH として設定され、ドライブ モードがプッシュプルになります。H/L ビットを「0」にセットすると、INT 出力ドライブはアクティブ LOW のオープン ドレイン出力として設定されます。したがって、デバイスによって駆動されていない場合、INT 出力を論理 HIGH 状態に駆動するための外部プルアップ抵抗が必要です。アクティブ LOW モード (H/L ビットが「0」にクリアされる) で INT を使用する場合、1kΩ～10kΩ の外部プルアップ抵抗を使用して INT ピンを V_{CC} にプルする必要があります。

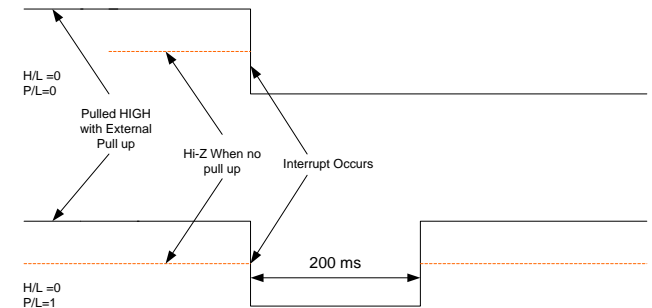
異なる H/L と P/L 設定に対応した INT ピンの動作: H/L ビットの設定は、割り込みが発生した時に INT ピン出力の状態が HIGH であるか、LOW であるかを決定します。同様に、P/L 設定は INT ピンのパルスまたはレベルを決定します。I²C nvSRAM の割り込みピン (INT) の動作は図 7 に示します。

図 7. INT ピンの動作 (RTC)

「1」にセットされた H/L



「0」にセットされた H/L



V_{RTCbat} と V_{RTCcap} ピン: システムの電源供給 (V_{CC}) がダウンしている場合、これらのピンは、RTC 回路網にバックアップ電源を供給するのに使用されて発振器のクロック実行を継続させます。電源喪失中の RTC 発振をバックアップするためには、V_{RTCbat} を再充電できないバッテリーに接続するか、V_{RTCcap} ピンのスーパーキャパシタに接続するかのどちらかです。使用されない場合、これらのピンはフローティングのままにする必要があります。

注意 V_{RTCcap} ピンは通常の動作中にそれに接続されているスーパー キャパシタを充電するために使用されるため、このピンを直接 V_{SS} に短絡することはできません。そのため、V_{RTCcap} ピンを直接グラウンド (V_{SS}) に接続すると、nvSRAM から過剰な電流を引き込むことがあります。

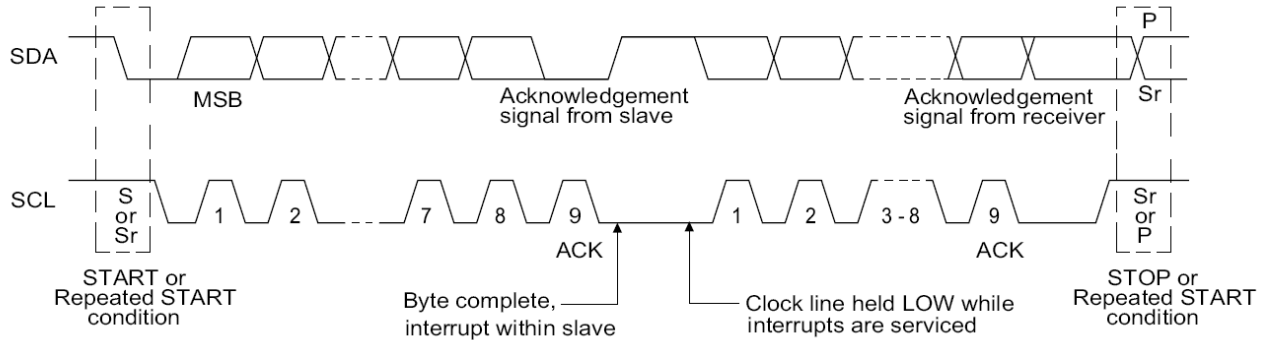
nvSRAM RTC の設計ガイドラインおよびベスト プラクティスについては、アプリケーション ノート、[AN61546 - Non Volatile Static Random Access Memory \(nvSRAM\) Real Time Clock \(RTC\) Design Guidelines and Best Practice](#) を参照してください。

I²C nvSRAM の動作

I²C nvSRAM のアクセスは常にバイト形式であり、SDA ライン上の全てのバイトは 8 ビット長でなければなりません。転送ごとに伝送できるバイト数は無制限のため、バースト モードの読

み書きに対応しています。各バイトは、ACK 応答 (A) ビットの後に続く必要があります。データは各バイト転送で最上位ビット (MSb) を最初に、最下位ビット (LSb) を最後に転送します。図 8 は I²C nvSRAM のデータ転送を示します。

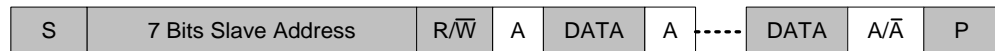
図 8. I²C nvSRAM のデータ転送



I²C nvSRAM のデータ転送は図 9 図 9 に示されている形式に従います。START 状態 (S) の後、スレーブ アドレスが送信されます。このアドレスはデータ方向ビット (R/W) である 8 番目のビット後に続く 7 ビット長です。ビット (R/W) が「0」にセットされると、転送 (WRITE) を示し、ビット (R/W) が「1」にセットされると、データのリクエスト (READ) を示します。データ転送は、常

にマスターで生成された STOP 状態 (P) で終了します。ただし、マスターがバスでの通信をまだ要求している場合は、反復の START 状態 (Sr) を生成し、スレーブを再びアドレスするか、または STOP 状態を生成せずに別のスレーブ デバイスで通信します。高速モードを除き、全ての標準的な I²C モードは図 9 に示すデータ形式に従います。

図 9. I²C のデータ バイト形式 (Sm、Fm、Fm+)



 = From I2C Master to I2C Slave

A = Acknowledged (SDA LOW during ACK clock)

 = From I2C Slave to I2C Master

A = Not Acknowledged (NACK) (SDA HIGH in ACK clock)

S/P = START /STOP Condition

図 10. データ バイト形式 (Sm、Fm、Fm+) – 書き込み処理

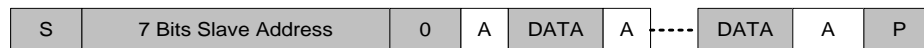


図 11. データ バイト形式 (Sm、Fm、Fm+) – 読み出し処理



高速モード (Hs モード) の動作

Hs モードでは、nvSRAM は最高 3.4Mbit/s までのビットレートでデータを転送することができます。START 状態(S) が生成された後、8 ビットのマスター コード (0000 1XXXb) は、nvSRAM が NACK (\bar{A})を送信しますが、全ての後続動作に対応した HS モードのデータ インターフェースを配置するために送信されます。デバイスは、STOP (P) 状態に従った後にのみ HS モードを終了します。スレーブを HS モードに配置した後、I²C マスターは、データ方向ビット (R/ \bar{W}) である 8 番目のビット

後に続く 7 ビットのスレーブ アドレスを転送します。ビット (R/ \bar{W})が「0」にセットされると、転送 (WRITE) を示し、ビット (R/ \bar{W}) が「1」にセットされると、データのリクエスト (READ) を示します。データ転送は、常にマスターで生成された STOP 状態 (P) で終了します。ただし、マスターが HS モードのバスで通信をまだ要求している場合は、反復の START 状態(Sr) を生成し、STOP 状態を生成せずにスレーブをアドレスすることができます。

図 12. I²C のデータ バイト形式 (Hs)

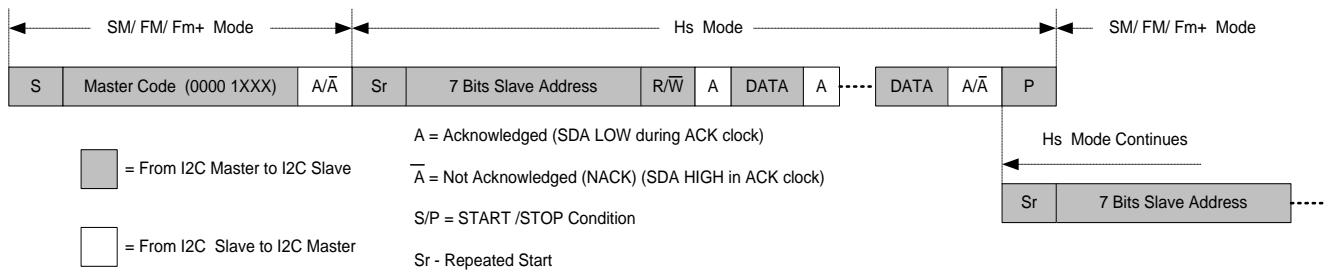


図 13. I²C のデータ バイト形式 (Hs) – 書き込み処理

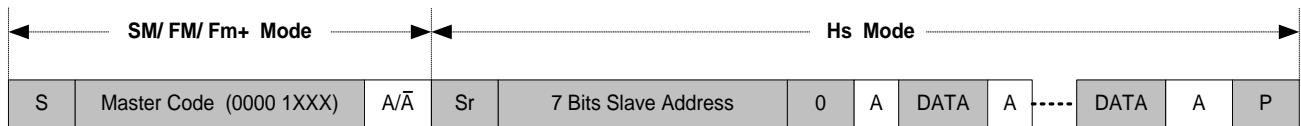
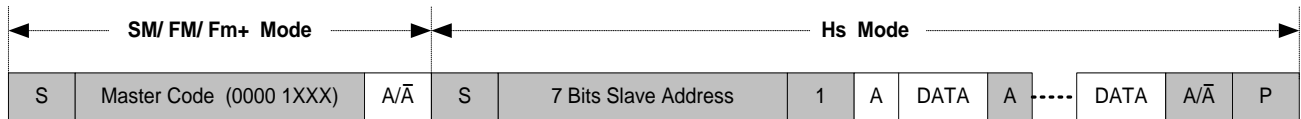


図 14. I²C のデータ バイト形式 (Hs) – 読み出し処理



I²C nvSRAM でのアドレス指定

I²C マスター コントローラは、バイト単位で I²C nvSRAM のスレーブと通信し、バイト転送の間は常に最初のクロック サイクルで最上位ビットを、8 番目のクロック サイクルで最下位ビットを転送します。これは、コマンド、アドレスおよびデータ バイトを含む全ての I²C 通信にとって良い状況です。同様に、I²C nvSRAM が読み出し処理中にデータ バイトを転送する場合は、常に最上位ビットを最初に、最下位ビットを最後に転送します。

図 15 は、I²C バスを介して転送されているアドレス ビット例を示します。

7 ビット長のスレーブ アドレスは、頭文字「SA [16:0]」を使用するメモリ アドレス ビットと区別するために、頭文字「SA [6:0]」で表現します。このため、これ以降の全ての節で、スレーブ アドレス ビットを SA [x]として表示します。

図 15. I²C nvSRAM でのアドレス ビット転送



スレーブ デバイス アドレス

I²C の nvSRAM スレーブは、7 ビットのスレーブ アドレス指定 SA [6:0] に対応しています。その内最上位の 4 アドレス ビット SA [6:3] がデバイス内で固定され、ユーザーは変更できません。残りの最下位 3 アドレス ビット SA [2:0] は、デバイスに用意されている外部アドレス ピン (A2、A1、A0) を介して設定可能です。I²C nvSRAM は、単一のデバイス内にデータ メモリ、

RTC 機能およびその他の制御として 3 つの異なる機能を備えています。I²C nvSRAM は、I²C マスターがこれらの機能にアクセスできるように、スレーブ アドレス SA [6:3] の上位 4 ビット (表 4 を参照) を固定することにより、3 つのユニークなスレーブ ID を割り当てています。

表 4. スレーブ デバイス アドレス

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	nvSRAM Function Select
1	0	1	0	Device select ID		A16/SA0/X	R/W	Selects Memory
1	1	0	1	Device select ID		X	R/W	Selects RTC Registers
0	0	1	1	Device select ID		X	R/W	Selects Control Registers

CY14X101I Slave Devices

Memory, 128 K × 8

RTC Registers, 16 × 8

Control Registers

- Memory Control Register, 1 × 8
- Serial Number, 8 × 8
- Device ID, 4 × 8
- Command Register, 1 × 8

バス上で他の任意のスレーブ ID が、上位 4 スレーブ アドレス ビット SA [6:3] に一致している場合、ユーザーは、ユニークなスレーブ ID が同じシステム バスを共有する各スレーブ デバイスに割り当てられるように、下位スレーブ アドレス ビット SA [2:0] を別々に設定する必要があります。

図 16. スレーブ デバイス アドレス選択

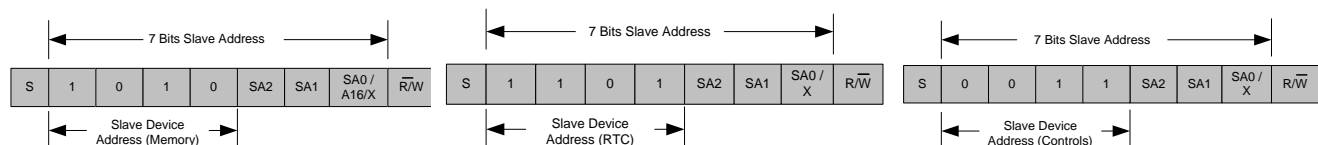


表 5. SRAM 読み書き用の I²C nvSRAM のアドレス指定

Density	Slave Address Byte								Address Byte2 (MSB)								Address Byte1 (LSB)											
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0				
64 Kbit	Slave Device Address								SA2	SA1	(SA0) ^{Note3}	R/W	(X) ^{Note2}	(X) ^{Note2}	(X) ^{Note2}	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
256 Kbit	Slave Device Address								SA2	SA1	(SA0) ^{Note3}	R/W	(X) ^{Note2}	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
512 Kbit	Slave Device Address								SA2	SA1	(SA0) ^{Note3}	R/W	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1 Mbit	Slave Device Address								SA2	SA1	A16	R/W	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

注: 2 最上位アドレス バイト (MSB) の未使用ビットは「ドントケア」ビットであり、nvSRAM によって無視されます。ただし、ファームウェア内で未使用のアドレス ビット位置を「0」にセットするのは良い方法です。このアプローチにより、将来のさらなる大容量オプションに移行する際に、ファームウェアを更新するのは容易になります。

注: 3 いくつかの nvSRAM デバイスのコンフィギュレーションでは、パッケージ上にピンが足りないか、または A0 アドレス ビットが内部で使用されているかのどちらかに起因して、2 個のアドレス ピン (A2 と A1) のみが用意されています。1M ビットのメモリ容量を持つ I²C nvSRAM は、その全体のメモリ位置をマッピングするために、17 アドレス ビット A [16:0] を必要とします。したがって、スレーブ アドレス空間 SA0 は、これらの部品で A16 アドレス ビットを転送するために使用され、2 ビット SA [2:1] のみがスレーブ アドレスを外部で設定することができます。ピン不足のパッケージ故に A0 が利用できない低メモリ容量のデバイス (512K ビット以下) では、このビットは内部でドントケア (「X」) になります。スレーブ アドレス ビット A0 がドントケアである I²C のスレーブ デバイスは、I²C のマスターによって送信された 2 つのスレーブ アドレス (A0=0 と A0=1) に対応して ACK 応答をします。

I²C nvSRAM のアクセス

標準的な機能 (メモリの読み書き) と特別な機能 (NV 処理、デバイス ID、シリアル番号) を含めた全ての I²C nvSRAM 機能は、標準的な I²C の読み書きプロトコルを介してアクセスされます。図 17～図 21 は、I²C nvSRAM の読み書き処理を説明

する簡略化したフロー図を示します。I²C nvSRAM の各機能とそれらの実装の詳細については、デバイスのデータシートを参照してください。

図 17. I²C nvSRAM のデータ メモリ書き込みの簡略フロー図

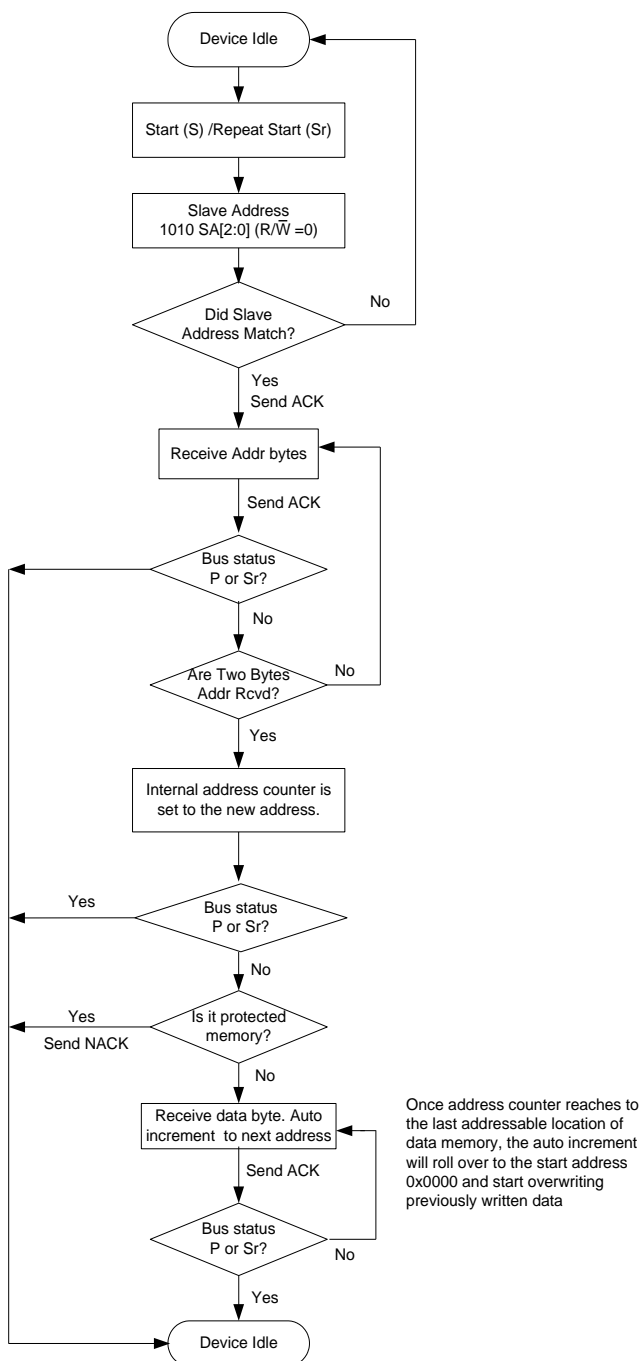


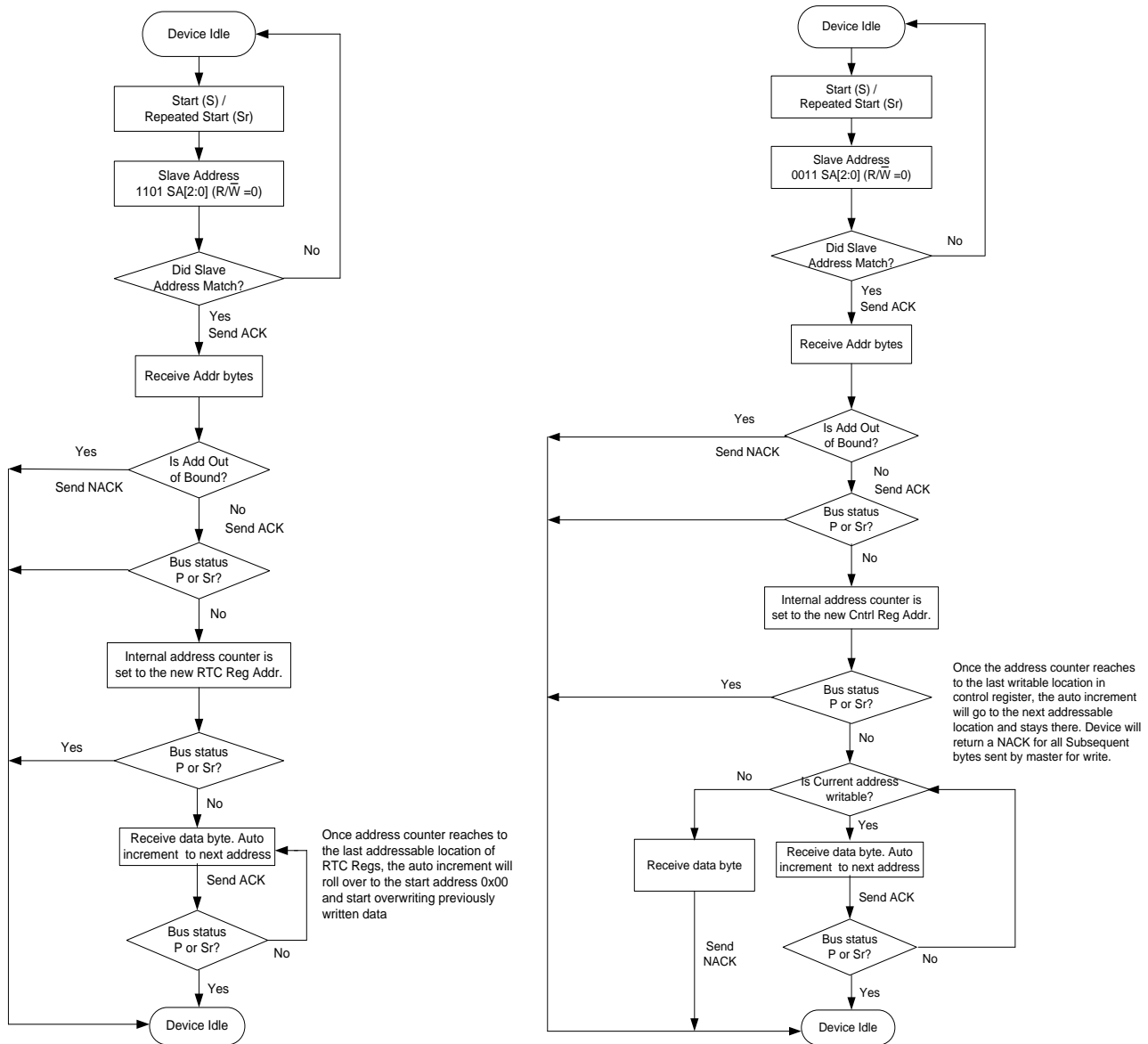
図 18. I²C nvSRAM RTC と制御レジスタ書き込みの簡略フロー図


図 19. I²C nvSRAM データ メモリの現時点でのアドレス読み出しの簡略フロー図

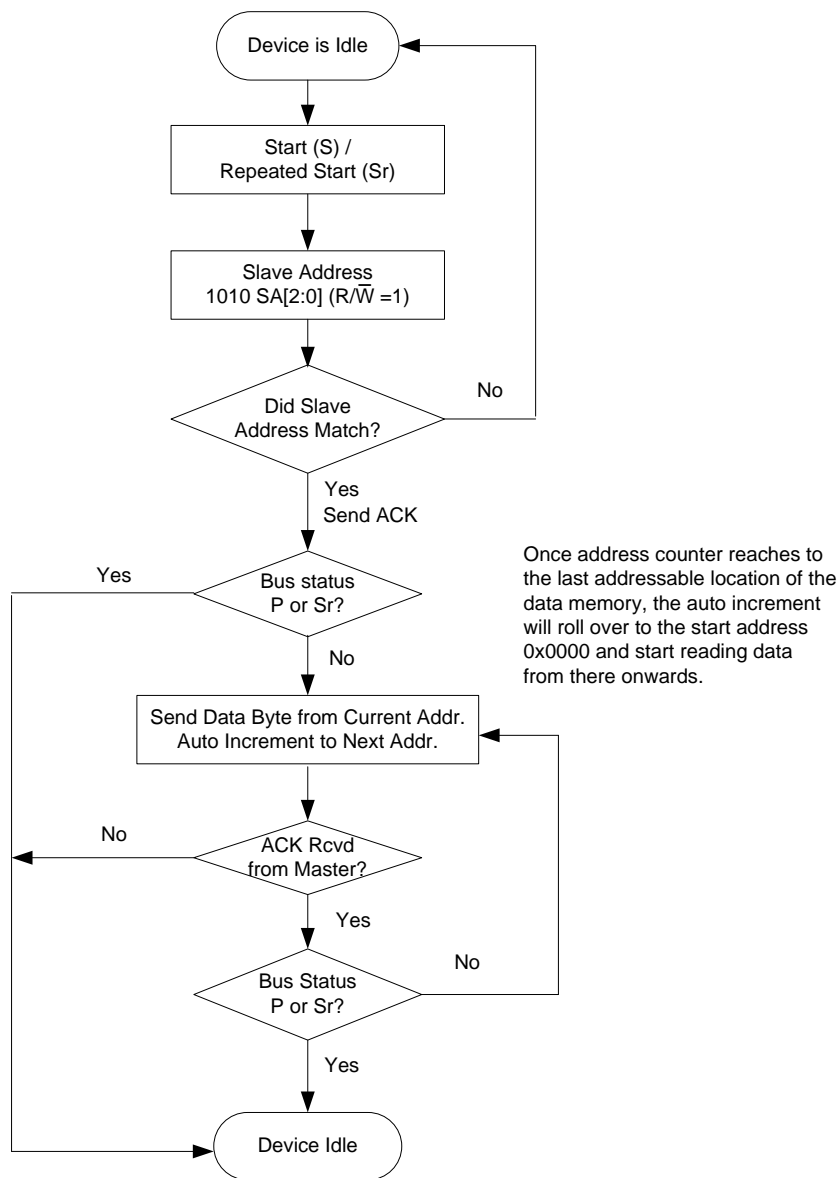


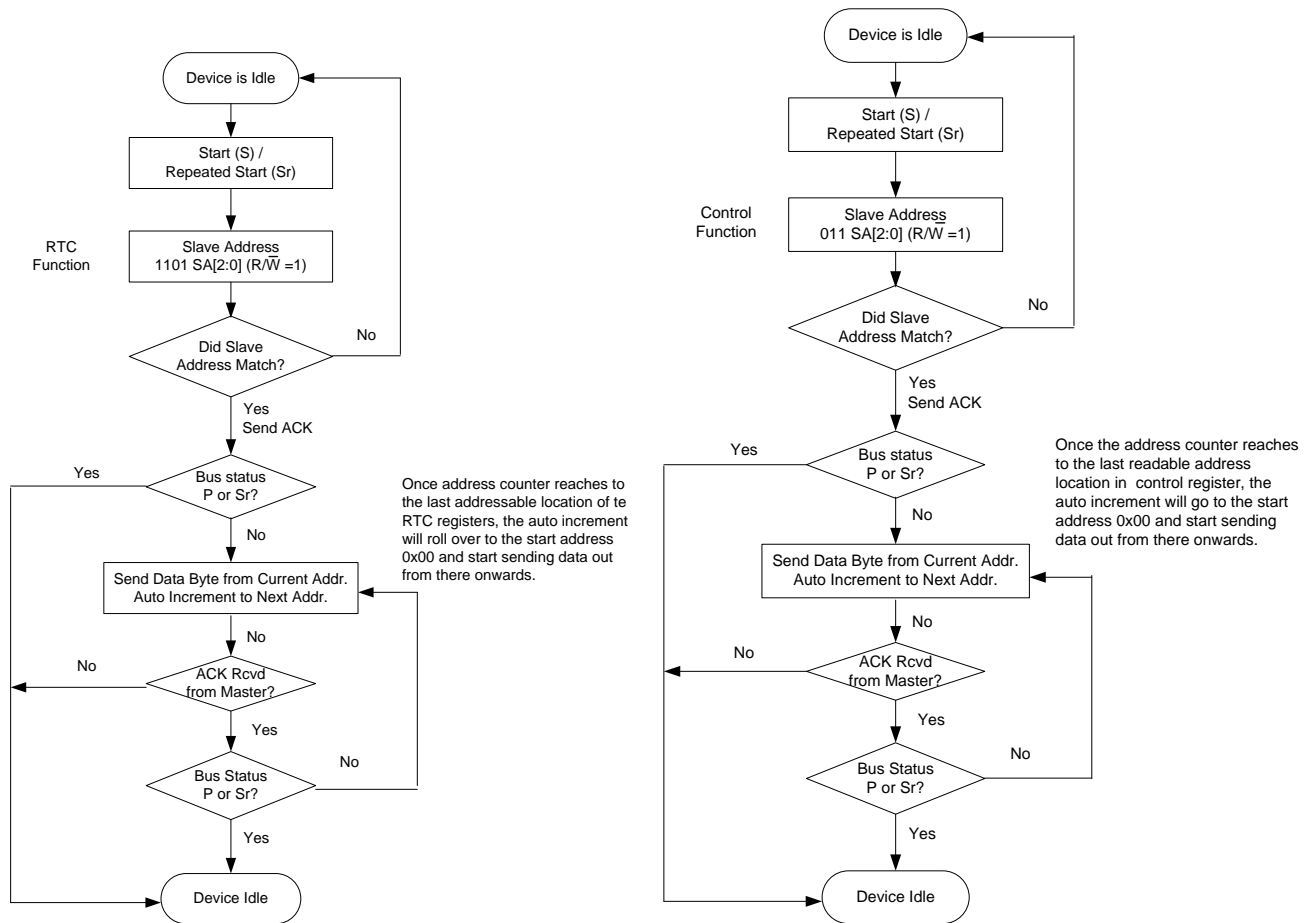
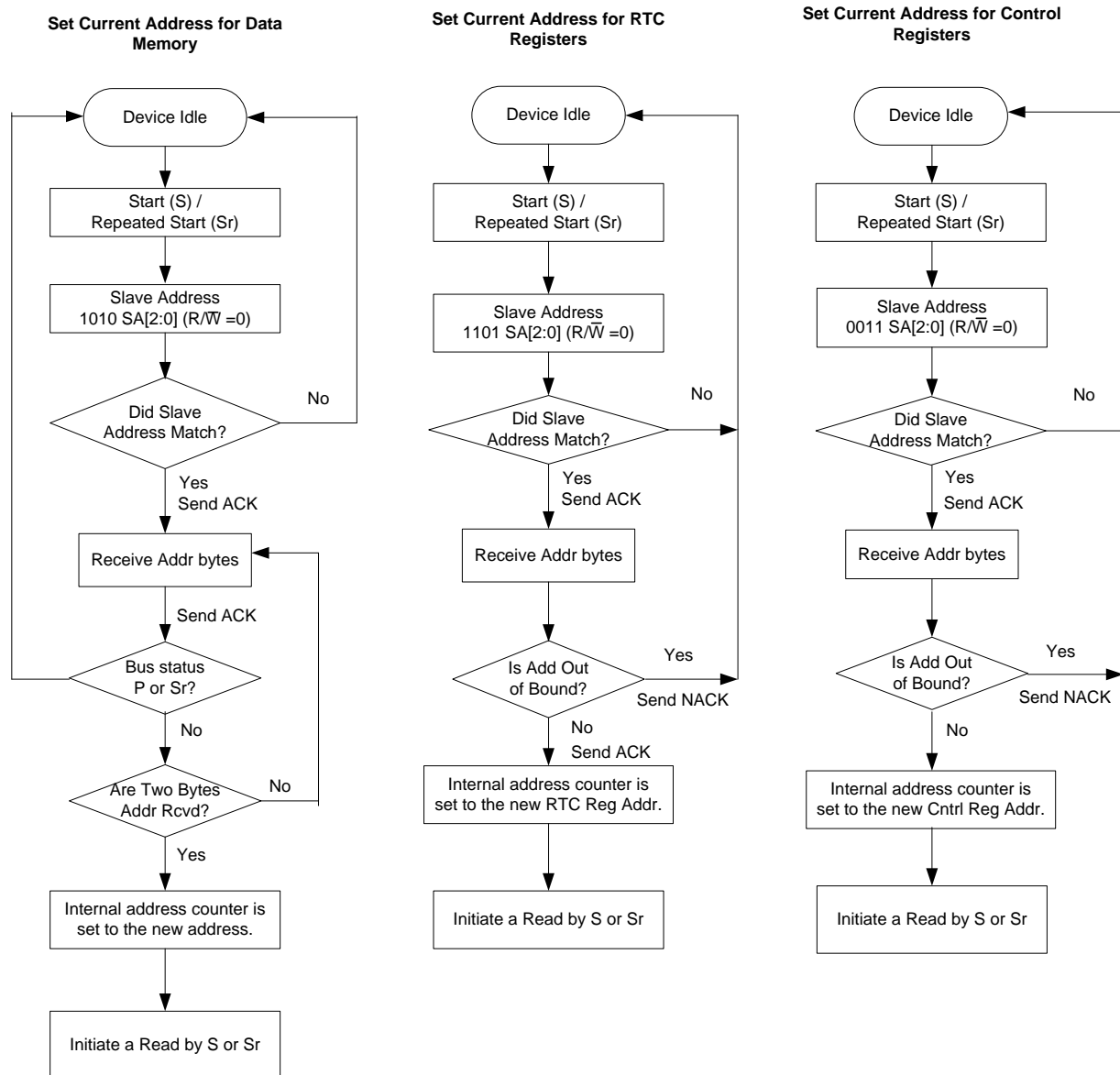
図 20. I²C nvSRAM RTC と制御レジスタの現時点のアドレス読み出しの簡略フロー図


図 19 と図 20 に示したフローチャートは、データ メモリ、RTC レジスタおよび制御レジスタから読み出された現時点の位置です。現時点の位置は、前の読み書き処理を終了する際のアドレス カウンタのアドレスです。ユーザーは別の位置から読み出しをしようとする場合、図 21 に示すように、書き込みサイクルを実行することでアドレス カウンタを新しいアドレスに設定する必要があります。

図 21. I²C nvSRAM データ メモリ、RTC、および制御レジスタのランダム アドレス読み出しの簡略フロー図


まとめ

サイプレスの I²C nvSRAM は、他の全ての不揮発性 I²C メモリ製品と同様に、標準的な I²C のアクセス プロトコルをサポートしています。これにより、nvSRAM と I²C マスター コントローラとの互換性が高まり、システム開発サイクル時間を低減します。このアプリケーション ノートは、回路図とタイミング図の助

けを借りてアプリケーションで I²C nvSRAM を設定する方法を示しています。

付録 A (疑似コード例)

I²C の書き込み

```
/*Sm, Fm, Fm+ Mode*/

void I2C_Write_nvSRAM(BYTE slave_Addr, BYTE Addr_MSB, BYTE Addr_LSB, BYTE *Data, int
n_Byte)
{
    int i=0;
    BYTE txBuffer[2];

    txBuffer[0]=Addr_MSB; //Copy I2C slave address in local buffer      txBuffer[1]=Addr_LSB;
    I2CHW_ClrWrStatus(); //Clear the status register of I2C master
    I2CHW_fSendStart(slave_Addr, I2CHW_WRITE); //Returns a non zero if slave device ACKs
        while(!I2CHW_bReadI2CStatus() & I2CHW_WR_COMPLETE); //Wait till all bits are
transmitted

    for (i=0; i<n_Byte; i++){
        I2CHW_fWrite( Data[i]); //Master transmit data bytes
            while (!I2CHW_bReadI2CStatus() & I2CHW_WR_COMPLETE);
    }
    I2CHW_SendStop (); // Master sends S/Sr to terminate write
\\}
```

```
/*Hs Mode*/

void I2C_Write_HSMODE_nvSRAM(BYTE slave_Addr, BYTE Addr_MSB, BYTE Addr_LSB, BYTE *Data,
int n_Byte)
\\{
    int i=0;
    BYTE txBuffer[2];

    txBuffer[0]=Addr_MSB;
    txBuffer[1]=Addr_LSB; //Copy I2C slave address in local buffer
    I2CHW_ClrWrStatus(); //Clear the status register of I2C master
    //0x00001xxx is a HS mode address hence. (Read/Write also don't care).HS mode command
byte can be set anything from 0x08 to 0x0F
    I2CHW_fSendStart( 0x04, I2CHW_WRITE); //No ACK from any slave.
        while(!I2CHW_bReadI2CStatus() & I2CHW_WR_COMPLETE); //Wait till all bits are
transmitted
    I2CHW_fSendRepeatStart(slave_Addr, I2CHW_WRITE); //Send repeat start with slave ID
to access a slave in HS mode.
        while(!I2CHW_bReadI2CStatus() & I2CHW_WR_COMPLETE);

    for (i=0; i<n_Byte; i++){
        I2CHW_fWrite( Data[i]);
            while (!I2CHW_bReadI2CStatus() & I2CHW_WR_COMPLETE);
            \\}
        I2CHW_SendStop (); //Master sends S/Sr to terminate write
    \\}
```

I²C の読み出し

```
/*Sm, Fm, Fm+ Mode*/
void I2C_Read_nvSRAM(BYTE slave_Addr, int n_Byte)
\\{
    int i=0;
    BYTE dataRD;

    I2CHW_ClrWrStatus(); //Clear the status register of I2C master
    I2CHW_fSendStart( slave_Addr, I2CHW_READ);
    while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE);

    for(i=0;i<n_Byte; i++) {
        if(i==(n_Byte-1)) {
            dataRD =I2CHW_bRead (I2CHW_NAKslave); //Master sends NACK for the last read to terminate
            the Read
            while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE); //Wait till all bits Rcvd
        }
        else {
            dataRD =I2CHW_bRead (I2CHW_ACKslave);
            while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE);
        }
    }
    I2CHW_SendStop (); //Master sends S/Sr to terminate Read
\\}
```

```
/*Hs Mode*/
void I2C_Read_HSMODE_nvSRAM(BYTE slave_Addr, int n_Byte)
\\{
    int i=0;
    BYTE dataRD;

    I2CHW_ClrWrStatus(); //Clear the status register of I2C master
    I2CHW_fSendStart( 0x04, I2CHW_READ); //0x0000 1xxx is a HS mode address hence slave addr
    can be 0x0X. No ACK from any slave.
    while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE); //Wait till all bits received
    I2CHW_fSendRepeatStart( slave_Addr, I2CHW_READ); //Send repeat start with slave ID to
    access a slave in the HS mode.
    while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE);

    for(i=0;i<n_Byte; i++){
        if(i==(n_Byte-1)) {
            dataRD =I2CHW_bRead (I2CHW_NAKslave);
            while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE);
        }
        else {
            dataRD =I2CHW_bRead (I2CHW_ACKslave);
            while(!I2CHW_bReadI2CStatus() & I2CHW_RD_COMPLETE);
        }
    }
    I2CHW_SendStop //Master sends S/Sr to terminate Read
\\}
```


改訂履歴

文書名: シリアル I²C nvSRAM を使った設計- AN74875

文書番号: 001-92723

版	ECN	変更者	提出者	変更内容
**	4395691	HZEN	06/17/2014	これは英語版 001-74875 Rev. *E を翻訳した日本語版 Rev. **です。

ワールドワイドな販売と設計サポート

サイプレスは、事業所、ソリューション センター、メーカー代理店および販売代理店の世界的なネットワークを保持しています。お客様の最寄りのオフィスについては、[サイプレスのロケーション ページ](#)をご覧ください。

製品

車載用	cypress.com/go/automotive
クロック&バッファ	cypress.com/go/clocks
インターフェース	cypress.com/go/automotive
照明&電源管理	cypress.com/go/powerpsoc cypress.com/go/plc
メモリ	cypress.com/go/memory
光学式ナビゲーションセンサ	cypress.com/go/ons
PSoC	cypress.com/go/psoc
タッチ センシング	cypress.com/go/touch
USB コントローラ	cypress.com/go/usb
ワイヤレス/RF	cypress.com/go/wireless

PSoC[®] ソリューション

psoc.cypress.com/solutions
PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

サイプレス開発者コミュニティ

[コミュニティ](#) | [フォーラム](#) | [ブログ](#) | [ビデオ](#) | [トレーニング](#)

テクニカル サポート

cypress.com/go/support

本書で言及するその他すべての商標または登録商標は、各社の所有物です。



Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

電話番号 : 408-943-2600
ファックス : 408-943-4730
ウェブサイト: www.cypress.com

© Cypress Semiconductor Corporation, 2011-2014. 本文書に記載される情報は、予告なく変更される場合があります。Cypress Semiconductor Corporation (サイプレス セミコンダクタ社) は、サイプレス製品に組み込まれた回路以外のいかなる回路を使用することに対して一切の責任を負いません。サイプレス セミコンダクタ社は、特許またはその他の権利に基づくライセンスを譲渡することも、または含意することはありません。サイプレス製品は、サイプレスとの書面による合意に基づくものでない限り、医療、生命維持、救命、重要な管理、または安全の用途のために使用することを保証するものではなく、また使用することを意図したものでもありません。さらにサイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことが合理的に予想される生命維持システムの重要なコンポーネントとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

このソースコード (ソフトウェアおよび/またはファームウェア) はサイプレス セミコンダクタ社 (以下「サイプレス」) が所有し、全世界の特許権保護 (米国およびその他の国)、米国の著作権法ならびに国際協定の条項により保護され、かつそれらに従います。サイプレスが本書面によりライセンシーに付与するライセンスは、個人的、非独占的かつ譲渡不能のライセンスであり、適用される契約で指定されたサイプレスの集積回路と併用されるライセンシーの製品のみをサポートするカスタム ソフトウェアおよび/またはカスタム ファームウェアを作成する目的に限って、サイプレスのソースコードの派生著作物をコピー、使用、変更して作成するためのライセンス、ならびにサイプレスのソースコードおよび派生著作物をコンパイルするためのライセンスです。上記で指定された場合を除き、サイプレスの書面による明示的な許可なくして本ソースコードを複製、変更、変換、コンパイル、または表示することは全て禁止します。

免責条項: サイプレスは、明示的または黙示的を問わず、本資料に関するいかなる種類の保証も行いません。これには、商品性または特定目的への適合性の黙示的な保証が含まれますが、これに限定されません。サイプレスは、本文書に記載される資料に対して今後予告なく変更を加える権利を留保します。サイプレスは、本文書に記載されるいかなる製品または回路を適用または使用したことによって生ずるいかなる責任も負いません。サイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことが合理的に予想される生命維持システムの重要なコンポーネントとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

ソフトウェアの使用は、適用されるサイプレス ソフトウェア ライセンス契約によって制限され、かつ制約される場合があります。