

M16Cファミリ、R8Cファミリ

RJJ05B1545-0102

 UARTi特殊モード1を使用したI²Cバスインタフェース

Rev.1.02

2010.12.20

1. 要約

この資料は、M16Cファミリ、R8Cファミリのシリアルインタフェース(UARTi)の特殊モード1(I²Cモード)を使用して、I²Cバスインタフェースを実現するための、UARTi特殊モード1の各機能および使用方法について説明しています。

2. はじめに

この資料で説明する応用例は、次のマイコンでの利用に適用されます。

- マイコン : M16Cファミリ、R8Cファミリ

この資料で使用している「i」(UARTi、UiMRレジスタなど)は、特殊モード1を使用できるシリアルインタフェースのチャンネルを示しています。特殊モード1を使用できるUARTiはマイコンにより異なりますので、ハードウェアマニュアルで確認してください。

UARTi特殊モード1によるI²Cバスインタフェースは、UARTiのクロック同期回路に付加されたI²Cバス通信のための補助機能を、ソフトウェアで制御することにより、簡易的にI²Cバス通信を実現します。通信制御をソフトウェアで行いますので、ハードウェアで実現するI²Cバスインタフェースモジュールに比べてソフトウェアの処理時間やタイミングに制約があります。お客様のソフトウェアにおける、I²Cバス通信以外のプログラムとの相互影響を含め、お客様システムでの十分な検証と評価を行ってください。

< タイトル別早見表 >

用途に応じて、該当する項目を参照してください。

タイトル	マスタ 送信	マスタ 受信	スレーブ 送信	スレーブ 受信
3. 初期設定				
3.1 I ² Cモードの設定				
3.2 SCLクロックの生成方法				
3.2.1 マスタ時のSCLクロック			—	—
3.2.2 スレーブ時のSCLクロック	—	—		
3.2.3 SCLクロックの数え方				
3.2.4 SCLクロックの周波数の考え方			—	—
3.3 その他のビットの設定				
3.3.1 IICM2ビットの使い方(割り込み要因の選択)				
3.3.2 STACビットの使い方(UARTi初期化機能)	—	—		
3.3.3 DL2 ~ DL0ビットの使い方(SDA デジタル遅延機能)				
3.3.4 CSCビットの使い方(クロック同期化機能)			—	—
4. コンディションの生成、検出				
4.1 各コンディションの生成(マスタ時)			—	—
4.2 スタートコンディション/ストップコンディションの検出(スレーブ時)	—	—		
4.3 CKPHビットの使い方(クロック遅延機能)				
4.4 SCLHIビットの使い方(SCL出力停止機能)			—	—
5. データ送受信(スレーブアドレス送信含む)				
5.1 バイトデータの送信方法				—
5.2 バイトデータの受信方法	—		—	
5.3 SWC、SWC9ビットの使い方(SCLウェイト機能)	—	—		
6. ACK / NACKの生成、検出				
6.1 ACK / NACKの生成方法	—			
6.2 ACK / NACKの検出方法				—
7. 割り込み				
7.1 スタートコンディション/ストップコンディション割り込み				
7.2 受信/送信割り込み				
7.2.1 受信割り込み	—	—		
7.2.2 送信割り込み				
8. UARTi特殊モード1(I ² Cモード)使用時の注意事項				
8.1 スレーブ送信/スレーブ受信時のUiTBレジスタのデータ設定手順	—	—		
8.2 電気的特性				
8.2.1 “L”レベル/“H”レベル入力電圧、“L”レベル出力電圧				
8.2.2 コンディション検出時のセットアップ時間およびホールド時間	—	—		
8.2.3 コンディション生成時のセットアップ時間およびホールド時間			—	—
8.3 UiBRGカウントソースによる最大転送速度の制限	—	—		
8.4 使用機能の制限				
8.4.1 UiSMR2レジスタのSWC2ビット(SCLウェイト機能)	—	—	—	—
8.4.2 UiSMR2レジスタのSDHIビット(SDA出力禁止機能)	—	—	—	—
8.4.3 スレーブ時のリスタートコンディション	—	—		

: 該当

—: 非該当

3. 初期設定

3.1 I²Cモードの設定

UiSMRレジスタのIICMビットを“1”(I²Cモード)にした後、UiMRレジスタのSMD2～SMD0ビットを“010b”(I²Cモード)にするとI²Cモードになります。

3.2 SCLクロックの生成方法

3.2.1 マスタ時のSCLクロック

マスタとして使用する場合、まずSCLクロックの転送レートを設定してください。

SCLクロックは、UiTBレジスタにデータを書き込んだ後、SCLクロックの1.5サイクル以内にSCLi端子から出力されます。

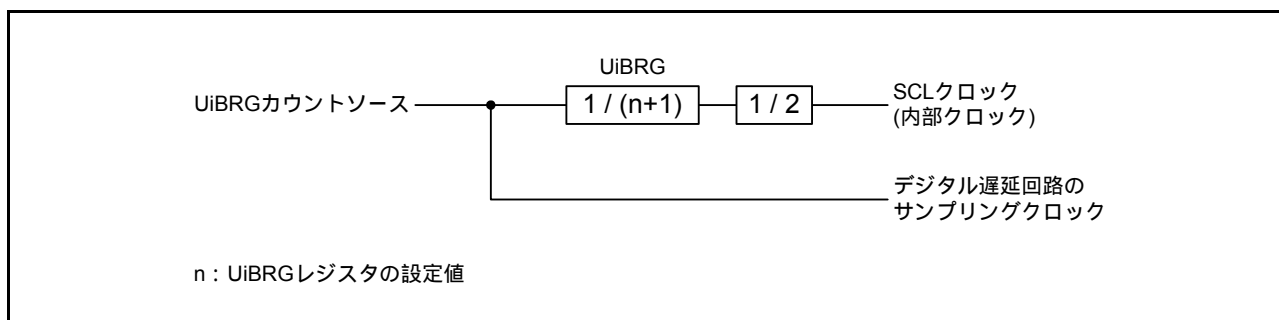


図 3.1 内部クロックの構成

3.2.2 スレーブ時のSCLクロック

スレーブとして使用する場合、UiMRレジスタのCKDIRビットを“1”(外部クロック)にしてください。そのとき、UiBRGレジスタの設定は無効になります。

ただし、UiBRGカウンタソースをデジタル遅延回路のサンプリングクロックとして使用するため、UiC0レジスタのCLK1～CLK0ビットを設定してください(「3.3.3 DL2～DL0ビットの使い方(SDAデジタル遅延機能)」参照)。

3.2.3 SCLクロックの数え方

本アプリケーションノートでの、SCLクロックの数え方を図 3.2に示します。

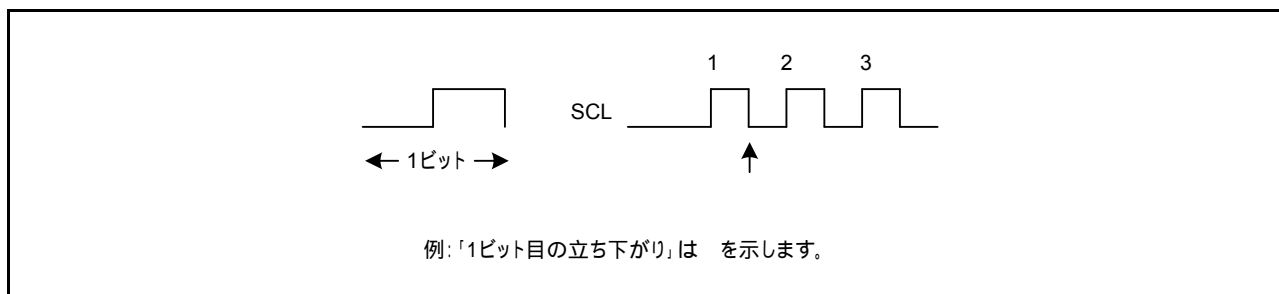


図 3.2 SCLクロックの数え方

3.2.4 SCLクロックの周波数の考え方

I²Cモードで生成するSCLクロックのDutyは50%です。そのため、I²CバスのFast-Modeの最大SCLクロック(400kbps)を設定すると、SCLクロックの“L”幅は1.25μsとなります。この値は、Fast-ModeのI²Cバス規格(f_{LOW} = 最小 1.3μs)を満たしません。よって、SCLクロックの設定を384.6kbps以下とし、SCLクロックの“L”幅が1.3μs以上になるようにしてください。

なお、クロック同期化機能(「3.3.4 CSCビットの使い方(クロック同期化機能)」参照)を有効にすると、ノイズフィルタ幅 + UiBRGカウンタソースの1 ~ 1.5サイクルのサンプリング遅延が発生し、SCLクロックの“H”認識が遅れるため、SCLクロックの“H”幅が延びます。そのため、SCLクロックの転送レートの設定に対して、実際のSCLクロックは遅くなります。

また、実際のSCLクロックを算出するためには、SCLクロック立ち上がり時間(t_R)も考慮する必要があります。

下記に実際のSCLクロックの算出例を示します。

< 384.6kbps設定時の実際のSCLクロックの算出例 >

- ・ UiBRGカウンタソース : $f_1 = 20\text{MHz}$
- ・ UiBRGレジスタの設定値 : $n = 26 - 1$
- ・ SCLクロック立ち上がり時間 : $t_R = 100\text{ns}$
- ・ SCLクロック立ち下がり時間 : $t_F = 0\text{ns}$
- ・ ノイズフィルタ幅 : $t_{NF} = 100\text{ns}$ (注1)
- ・ サンプリング遅延 : $t_{SD} = 1$ サイクル

の場合、

$$f_{SCL}(\text{理論値}) = f_1 / (2(n+1)) = 20\text{MHz} / (2(25+1)) = 384.6\text{kbps}$$

$$t_{LOW} = 1 / (2f_{SCL}(\text{理論値})) = 1 / (2 \times 384.6\text{kbps}) = 1.3\mu\text{s}$$

$$t_{HIGH} = 1 / (2f_{SCL}(\text{理論値})) + t_{NF} + (t_{SD} \times 1 / f_1)$$

$$= 1 / (2 \times 384.6\text{kbps}) + 100\text{ns} + (1 \times 1 / 20\text{MHz})$$

$$= 1.45\mu\text{s}$$

$$f_{SCL}(\text{実効値}) = 1 / (t_F + t_{LOW} + t_R + t_{HIGH}) = 1 / (0\text{ns} + 1.3\mu\text{s} + 100\text{ns} + 1.45\mu\text{s}) \quad 350.8\text{kbps}$$

注1. 最大200ns。

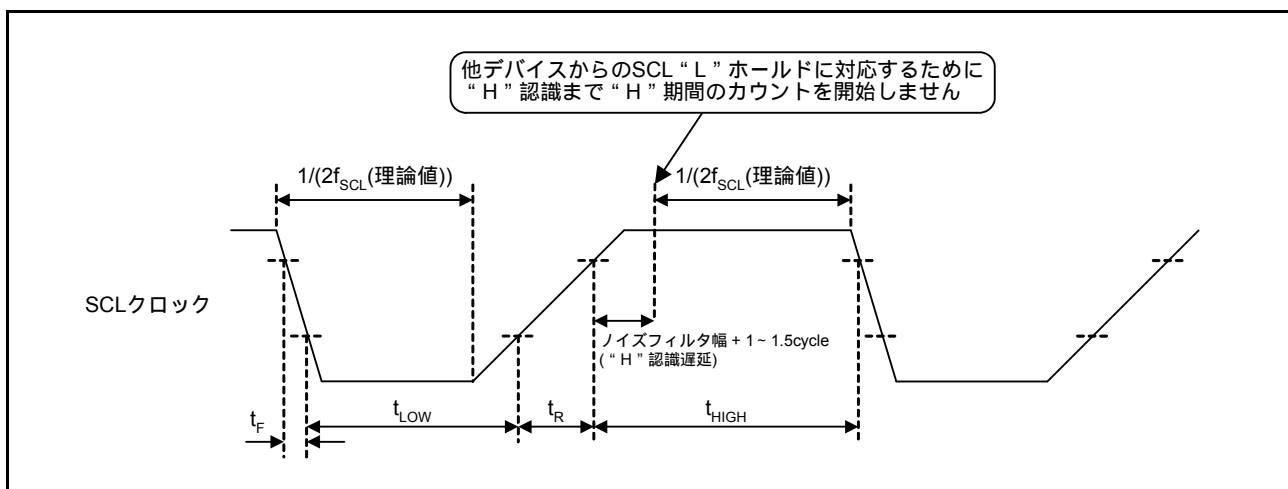


図 3.3 SCLクロック

3.3 その他のビットの設定

3.3.1 IICM2ビットの使い方(割り込み要因の選択)

UiSMR2レジスタのIICM2ビットは、I²Cモード時の割り込み要因を選択するビットです。通常、IICM2ビットは“1”(UART送信/UART受信割り込み)で使用してください。

3.3.2 STACビットの使い方(UARTi初期化機能)

UARTi初期化機能は、スタートコンディション検出時に自動的にUARTiを初期化する機能で、スレープ時に使用してください。UiSMR2レジスタのSTACビットを“1”(許可)にすると機能が許可され、“0”(禁止)にすると禁止されます。

スレープ時、UARTi初期化機能を使用すると、スタートコンディション検出時に自動的にUARTiが初期化されるため、スタートコンディション検出時に割り込みは必要ありません。

STACビットが“1”のとき、スタートコンディションを検出すると、以下の初期化が行われます。

- (1) 送信シフトレジスタは初期化され、UiTBレジスタの内容が送信シフトレジスタに転送されます。これにより、データ受信時にUiTBレジスタにデータを再設定する必要はなく、次に入力されたクロックを1ビット目として送信が開始されます。
ただし、そのときの送信データは最後に送信していたデータと同じものとなりますので、UiSMR4レジスタのACKDビットを“1”(NACK)、ACKCビットを“1”(ACKデータ出力)にして、送信データの出力を禁止してください。
- (2) 受信シフトレジスタは初期化され、次に入力されたクロックを1ビット目として、スレープアドレスの受信が開始されます。UiRBレジスタを読み出す前のタイミングで初期化され受信が開始しても、オーバランエラーは発生しません。
- (3) UiSMR2レジスタのSWCビットが自動的に“1”(許可)になります。これにより、SCLウェイト機能が有効になり、SCLクロックの8ビット目の立ち下がり、SCLi端子が“L”固定になります。

3.3.3 DL2 ~ DL0ビットの使い方(SDAデジタル遅延機能)

I²Cバスでデータ送信を行う場合、SCLクロックが“L”のときにデータを変化させる必要があります。SCLクロックが“H”のときにSDAが変化すると、各コンディションと認識されます(「8.2.3 コンディション生成時のセットアップ時間およびホールド時間」参照)。

SDAデジタル遅延機能は、SDAi端子からの出力を遅延させる機能です。SDAの変化を遅延させることで、SCLクロックが“L”でのデータ変化を確定できます。

SDAデジタル遅延機能は、UiSMR3レジスタのDL2 ~ DL0ビットを“001b” ~ “111b”にすると有効になり、“000b”にすると無効になります。

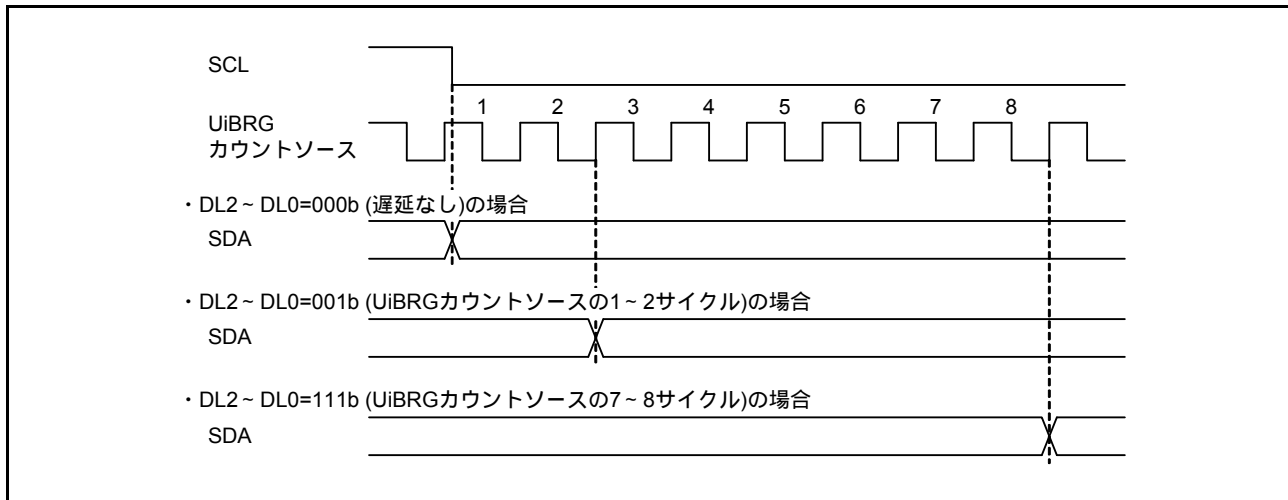


図 3.4 DL2 ~ DL0ビットの設定によるSDA出力切り替え

3.3.4 CSCビットの使い方(クロック同期化機能)

マスタ時は、UiSMR2レジスタのCSCビットを“1”(許可)にして、クロック同期化機能を許可してください。スレーブ時は、CSCビットは“0”(禁止)にしてください。

クロック同期化機能は、他デバイスからのSCLi端子への“L”ホールドに対して自動的に待ち状態に入り、“L”ホールドの解除によって待ち状態を解除する機能です。

クロック同期化機能を使用すると、SCLクロックの設定に対して、実際のSCLクロックは遅くなります。詳細は「3.2.4 SCLクロックの周波数の考え方」を参照してください。

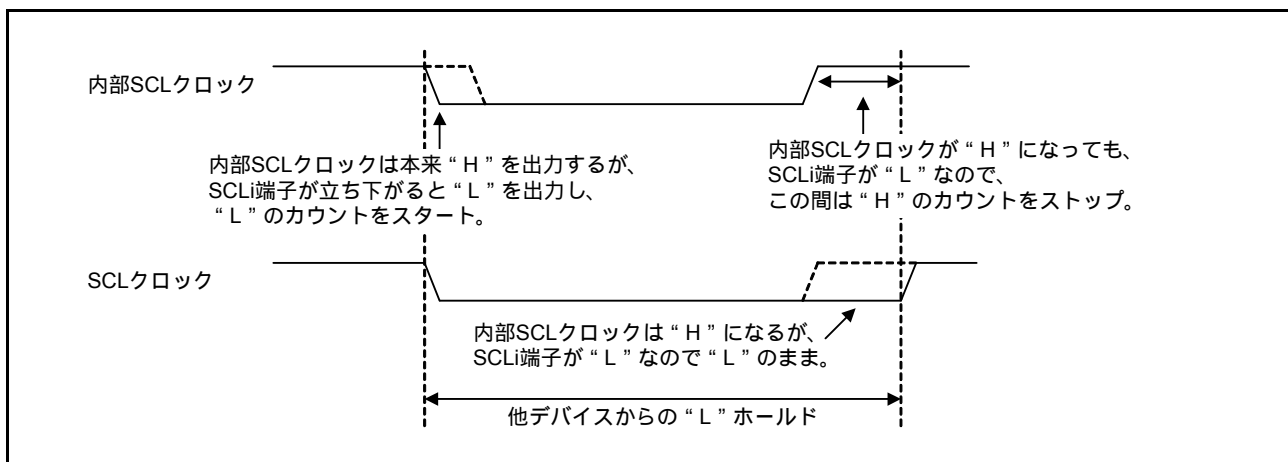


図 3.5 クロック同期化機能

4. コンディションの生成、検出

4.1 各コンディションの生成(マスタ時)

マスタ時のスタートコンディション、ストップコンディション、およびリスタートコンディションは、ハードウェアで生成できます。

UiSMR4レジスタのSTAREQビットを“1”(スタート)にすると、スタートコンディションを生成します。UiSMR4レジスタのSTPREQビットを“1”(スタート)にすると、(SCLi端子が“L”のときはSCLi端子の開放を待ってから)ストップコンディションを生成します。

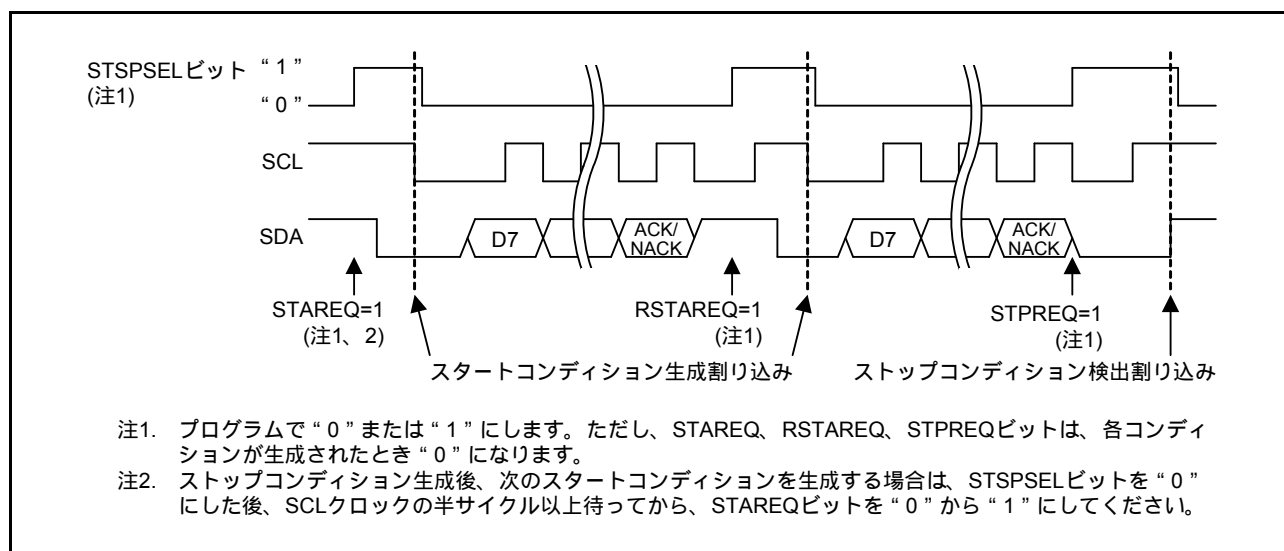
UiSMR4レジスタのRSTAREQビットを“1”(スタート)にすると、(SCLi端子が“L”のときはSCLi端子の開放を待ってから)リスタートコンディションを生成します。

STAREQビット、STPREQビット、RSTAREQビットは、各コンディションが生成されたとき、“0”になります。

UiSMR4レジスタのSTSPSELビットを“1”(コンディション出力する)にすると、上記ビットに対応した各コンディションを生成します。各コンディション生成時は、必ず上記のいずれかのREQビットを“1”にしてから、STSPSELビットを“1”にしてください。

図 4.2に各コンディション生成時のレジスタ設定手順を示します。

スタートコンディションおよびストップコンディション生成時のセットアップ時間、ホールド時間の詳細は「8.2.3 コンディション生成時のセットアップ時間およびホールド時間」を参照してください。



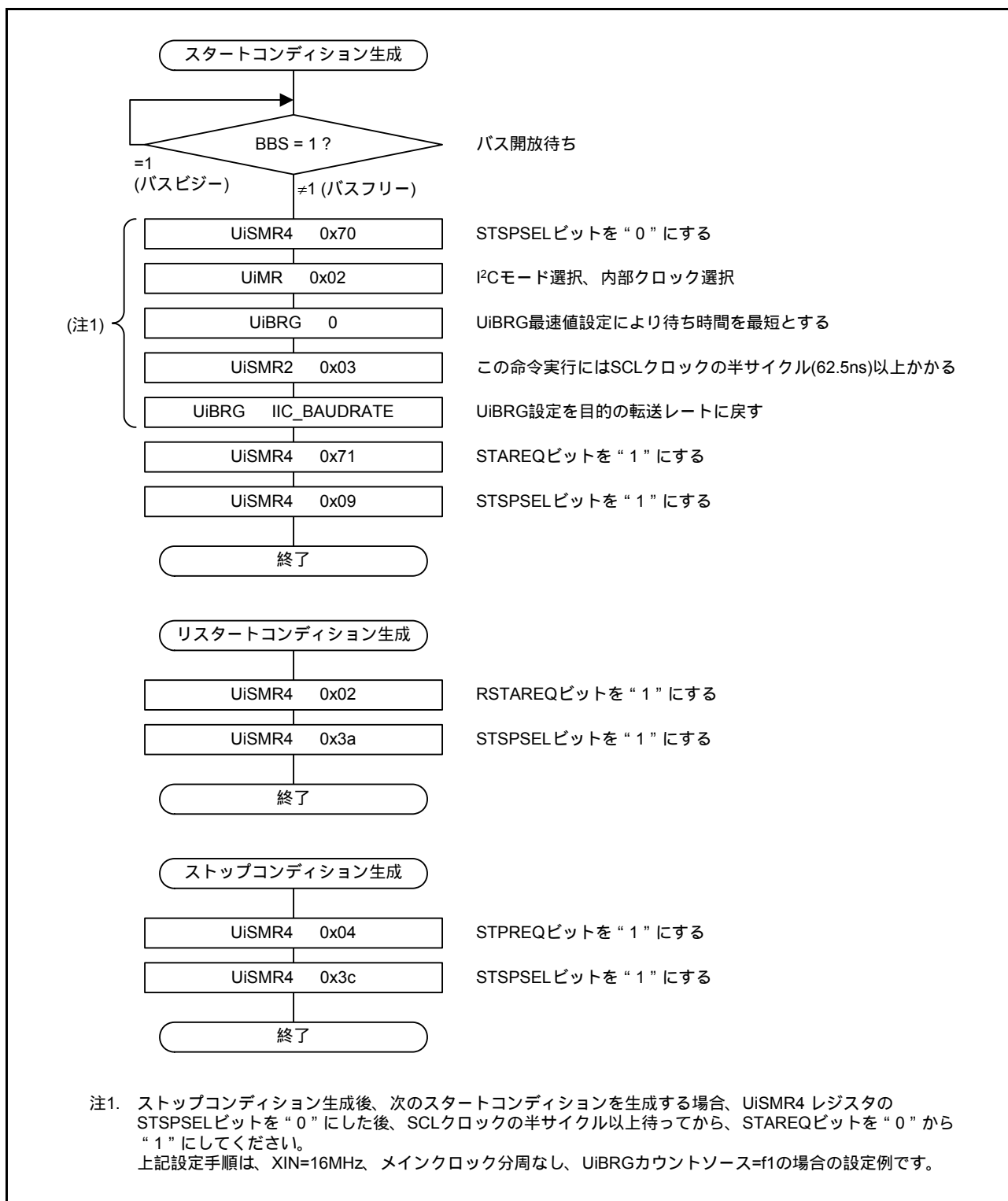


図 4.2 各コンディション生成時のレジスタ設定手順

4.2 スタートコンディション/ストップコンディションの検出(スレーブ時)

スレーブ時のスタートコンディションおよびストップコンディションは、スタートコンディション/ストップコンディション検出割り込みで検出できます。詳細は「7.1 スタートコンディション/ストップコンディション割り込み」を参照してください。

スタートコンディションおよびストップコンディション検出時のセットアップ時間、ホールド時間は、I²Cバス規格と異なる場合があります。詳細は「8.2.2 コンディション検出時のセットアップ時間およびホールド時間」を参照してください。

4.3 CKPHビットの使い方(クロック遅延機能)

I²Cモード時は、クロック遅延機能を使用してください。

クロック遅延機能は、UiSMR3レジスタのCKPHビットを“1”(クロック遅れあり)にすると有効になります。クロック遅延機能を使用すると、受信シフトレジスタからUiRBレジスタへの転送が2回行われます。それぞれのUiRBレジスタの内容は「図 4.3 クロック遅延機能」を参照してください。

マスタ時、CKPHビットはUiSMR4レジスタのSTSPSELビットが“1”(スタートコンディション、ストップコンディション出力する)のときに設定してください。スタートコンディション生成前はCKPHビットは“0”(クロック遅れなし)にしてください。スタートコンディション生成後、STSPSELビットを“0”(スタートコンディション、ストップコンディション出力しない)にする前にCKPHビットを“1”にして、ストップコンディション生成後、STSPSELビットを“0”にする前にCKPHビットを“0”にしてください。

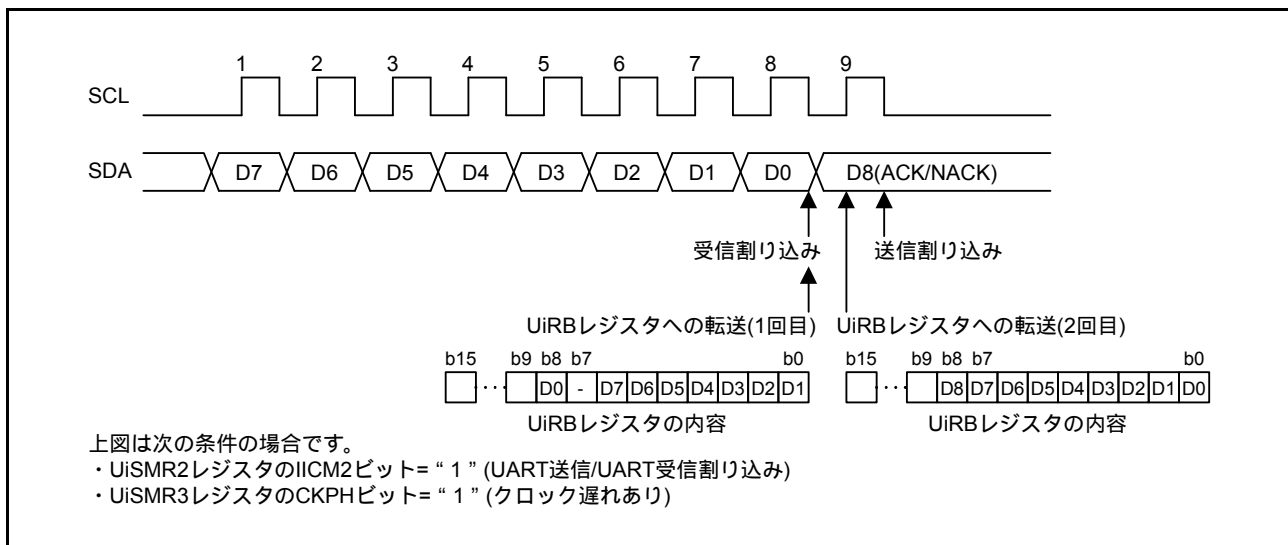


図 4.3 クロック遅延機能

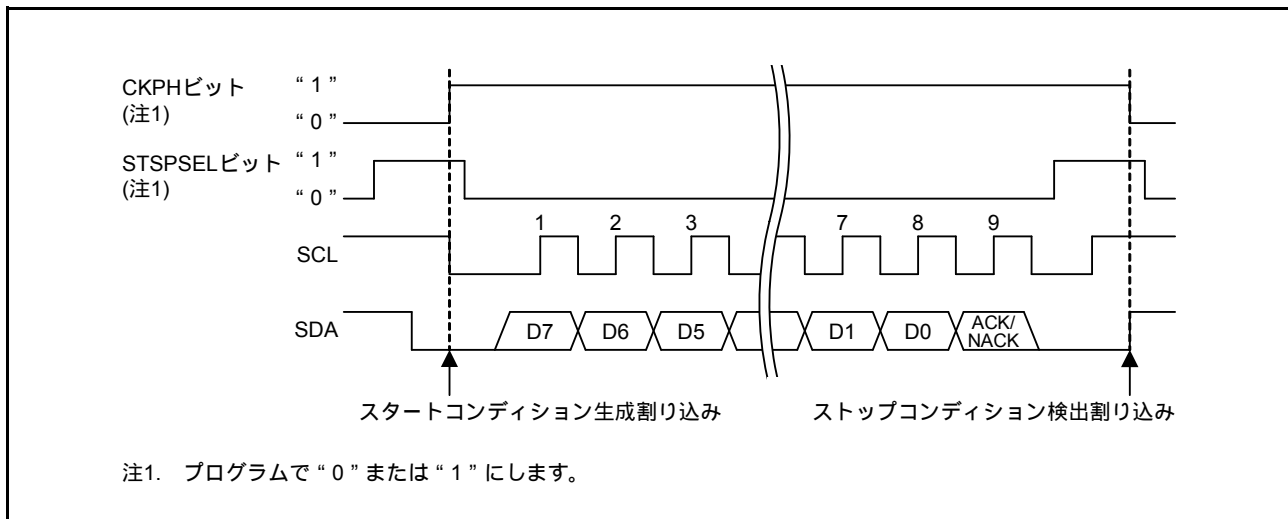


図 4.4 CKPHビットの設定(マスタ時)

4.4 SCLHIビットの使い方(SCL出力停止機能)

マスタ時は、スタートコンディション生成時に設定します。詳細は「図 4.2 各コンディション生成時のレジスタ設定手順」を参照してください。

スレープ時は、SCLHIビットを“0”(禁止)にしてください。

5. データ送受信(スレーブアドレス送信含む)

5.1 バイトデータの送信方法

バイトデータを送信する場合、1ビット目～8ビット目はSDAi端子から送信データを出力し、9ビット目はアクノリッジを受信するためSDAi端子を開放します。

I²Cモードでは、UiTBレジスタに9ビットのデータを設定することで、送信データの出力とSDAi端子の開放ができます。9ビットデータのb7～b0には送信データを、b8はSDAi端子を開放するため“1”を設定してください。

UiC0レジスタのUFORMビットを“1”(MSBファースト)にして、UiTBレジスタに9ビットデータを設定すると、b7 b6 … b0 b8の順で、SDAi端子からデータが出力されます。b8を“1”にすることで9ビット目のSDAi端子がハイインピーダンス状態になり、アクノリッジを受信できます。

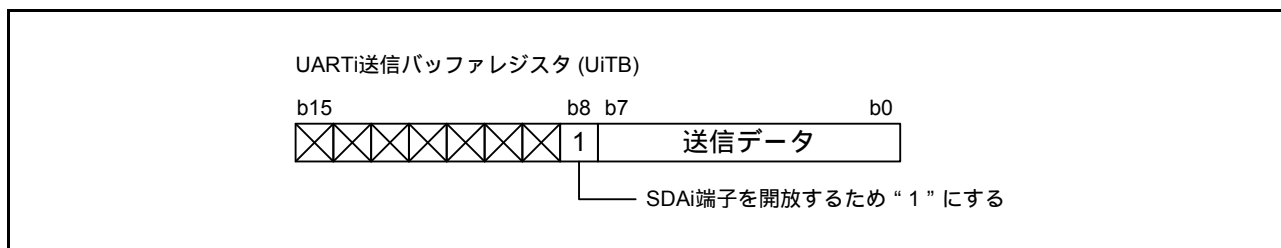


図 5.1 UiTBレジスタの設定

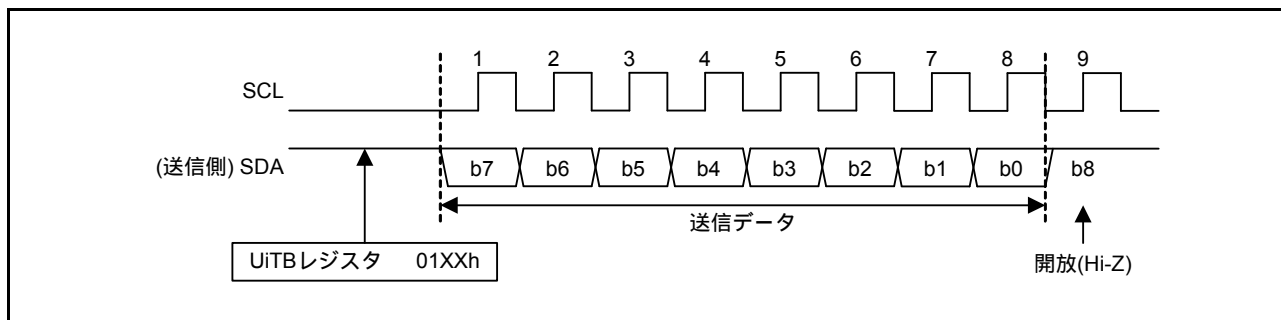


図 5.2 バイトデータの送信

5.2 バイトデータの受信方法

バイトデータを受信する場合、1ビット目～8ビット目はデータを受信するためSDAi端子を開放し、9ビット目はアクノリッジを生成します。アクノリッジは、マスタ時の最終バイトデータを受信するとき、またはスレーブ時のスレーブアドレス不一致のときはNACKを生成します。それ以外は、通常、ACKを生成します。

I²Cモードでは、UiTBレジスタに9ビットのデータを設定することで、データの受信およびアクノリッジの生成ができます。9ビットデータのb7～b0にはSDAi端子を開放するため“FFh”を、b8はACKを生成する場合は“0”、NACKを生成する場合は“1”を設定してください。

UiTBレジスタに9ビットデータの“00FFh”または“01FFh”を設定すると、1ビット目～8ビット目はSDAi端子がハイインピーダンス状態になり、データを受信できます。9ビット目は設定したACKまたはNACKが生成されます。

受信したデータはUiRBレジスタから読み出します。クロック遅延機能を使用すると、UiRBレジスタへのデータ転送が2回行われ、それぞれのUiRBレジスタの内容が異なります。詳細は「4.3 CKPHビットの使い方(クロック遅延機能)」を参照してください。

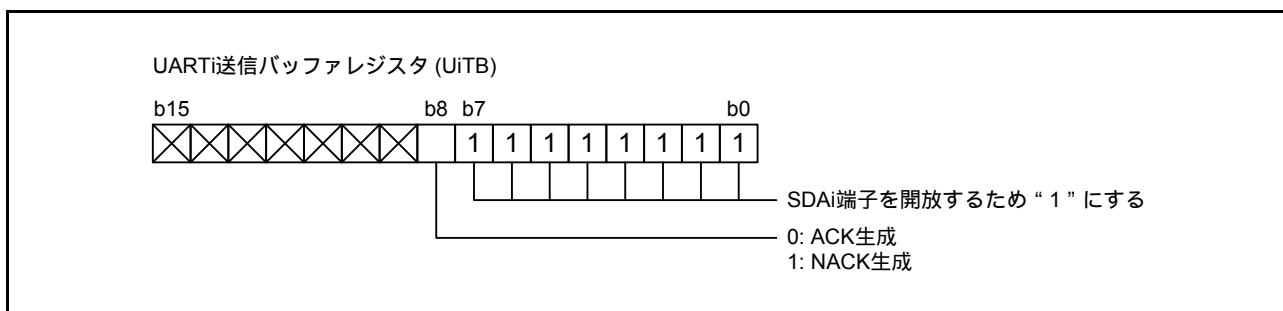


図 5.3 UiTBレジスタの設定

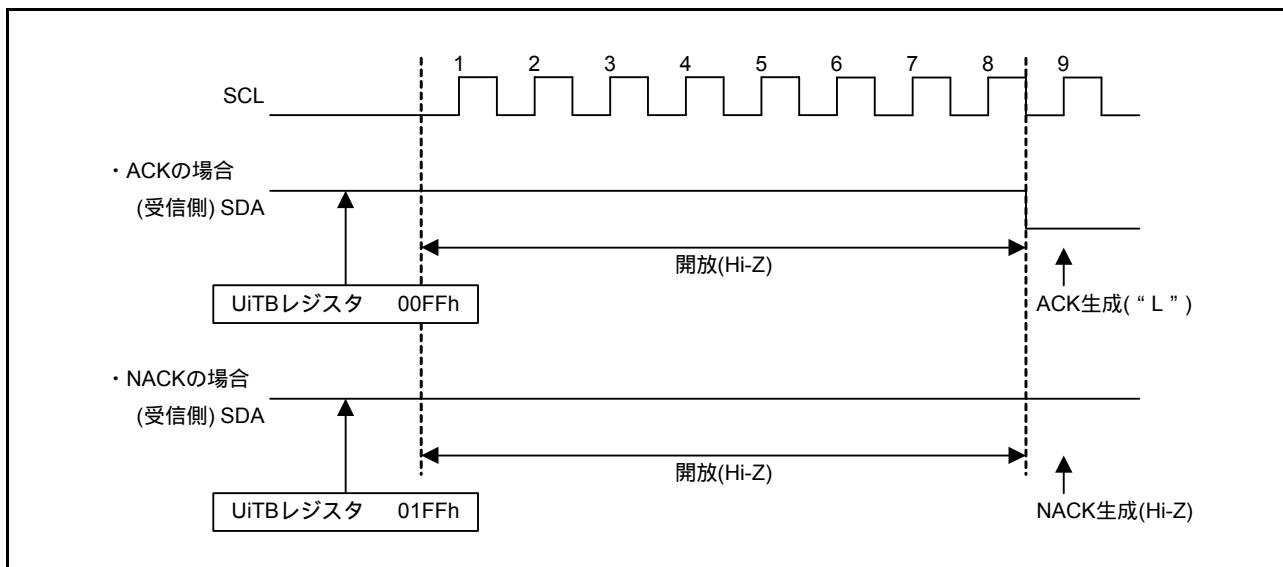


図 5.4 バイトデータの受信

5.3 SWC、SWC9ビットの使い方(SCLウェイト機能)

SCLウェイト機能は、アクノリッジの生成やデータの準備などの処理に必要な時間を確保するために強制的に他デバイスを待ち状態にする機能で、スレーブ時に使用します。

UiSMR2レジスタのSWCビットは、アクノリッジ生成のためのウェイトを挿入するときに使用します。SWCビットが“1”(8ビット受信後、SCLi端子を“L”に固定)の場合、SCLクロックの8ビット目の立ち下がり(SCLi端子が“L”固定になります)。SWCビットを“0”(禁止)にすると、SCLi端子の“L”固定を解除できます。

UiSMR4レジスタのSWC9ビットは、受信したアクノリッジを判定するためのウェイトを挿入するときに使用します。UiSMR3レジスタのCKPHビットが“1”(クロック遅れあり)のとき、SWC9ビットを“1”(9ビット受信後、SCLi端子を“L”に固定)にすると、SCLクロックの9ビット目の立ち下がり(SCLi端子が“L”固定になります)。SWC9ビットを“0”(SCL“L”ホールド禁止)にするとSCLi端子の“L”固定を解除できます。

受信/送信割り込みのタイミングは「7.2 受信/送信割り込み」を参照してください。

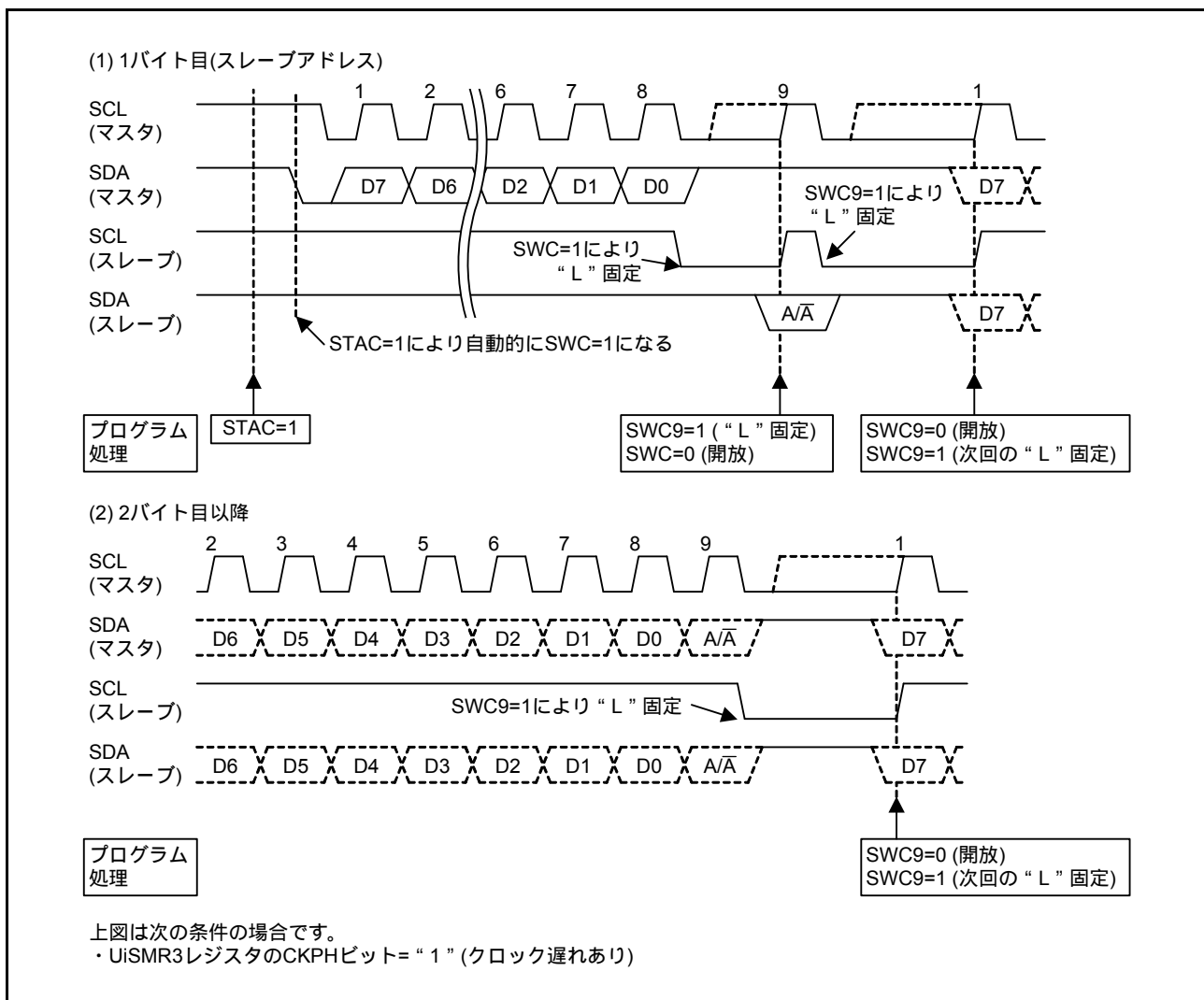


図 5.5 SCLウェイト機能

6. ACK / NACKの生成、検出

6.1 ACK / NACKの生成方法

データ(スレーブアドレスを含む)を受信する場合、受信側は9ビット目にアクノリッジを生成します。

UiSMR4レジスタのACKDビットとACKCビットでアクノリッジを生成できます。ACKCビットを“1”(ACKデータ出力)にすると、UiTBレジスタに設定した内容は出力されず、ACKDビットの設定内容が出力されます。ACKDビットが“0”(ACK)のときはSDAi端子が“L”になり、ACKDビットが“1”(NACK)のときはSDAi端子がハイインピーダンス状態になります。

データを受信してからACK/NACKを生成する場合、SCLクロックの8ビット目の立ち下がり、ACKDビットを“0”、ACKCビットを“1”にすると、ACKが生成されます。ACKDビットを“1”、ACKCビットを“1”にすると、NACKが生成されます。ACKCビットは、SCLクロックの9ビット目の立ち下がり、で、“0”(シリアルインタフェースデータ出力)にしてください(「図 6.1 ACKCビット、ACKDビットの設定」参照)。

データ受信開始時に、生成するACK/NACKが確定している場合は、UiTBレジスタに“00FFh”(ACK)または“01FFh”(NACK)を設定することでも、アクノリッジを生成できます(「5.2 バイトデータの受信方法」参照)。このとき、ACKCビットは“0”にしてください。

スレーブ時、スレーブアドレスの一致判定に従ってACK/NACKを生成するときは、受信割り込み処理でACKDビットに“0”または“1”を設定し、ACKCビットを“1”にしてください(「7.2 受信 / 送信割り込み」参照)。

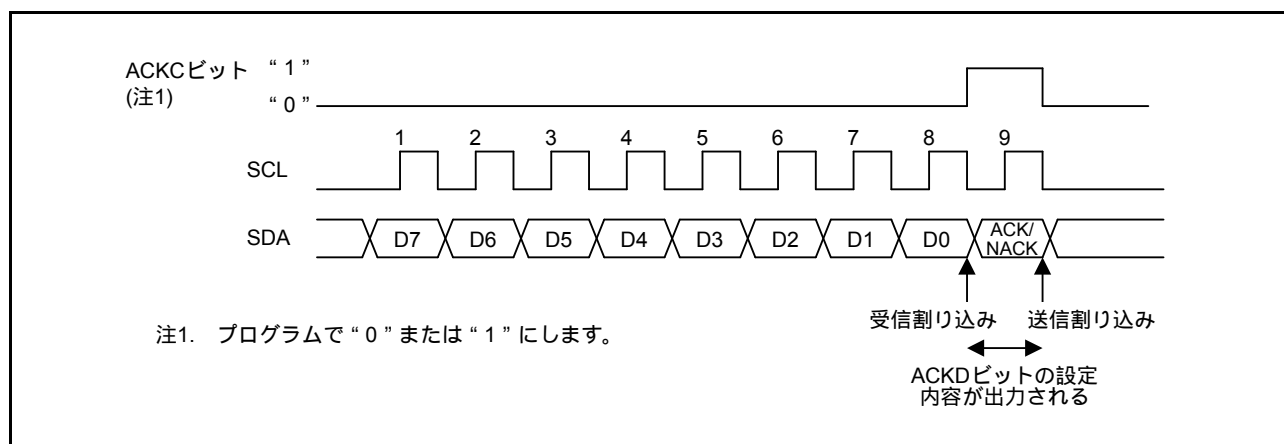


図 6.1 ACKCビット、ACKDビットの設定

6.2 ACK / NACKの検出方法

データ(スレーブアドレスを含む)を送信する場合、送信側は9ビット目にアクノリッジを受信します。送信割り込み処理でUiRBレジスタのb8を確認して、受信したアクノリッジを判定してください(「5.1 バイトデータの送信方法」参照)。

7. 割り込み

7.1 スタートコンディション/ストップコンディション割り込み

スタートコンディション割り込みは、UiSMR4レジスタのSTSPSELビットの設定によって、割り込み要求発生タイミングが異なります。

STSPSELビットが“0”(コンディション出力しない)の場合、スタートコンディション検出時およびストップコンディション検出時に割り込み要求が発生します。

STSPSELビットが“1”(コンディション出力する)の場合、スタートコンディション生成終了時およびストップコンディション検出時に割り込み要求が発生します。

マスタ時、STSPSELビットは、スタートコンディション/ストップコンディション生成割り込み処理で“0”にしてください。

スタートコンディションのSDAの立ち下がり、ストップコンディションのSDAの立ち上がりでUiSMRレジスタのBBSビットが“1”(スタートコンディション検出)になります。ストップコンディションのSDAの立ち上がりでBBSビットが“0”(ストップコンディション検出)になります。割り込み要求発生時にBBSビットを判定して、スタートコンディションまたはストップコンディションの処理を行ってください。

スレープ時、UARTi初期化機能を使用すると、スタートコンディション検出時に自動的にUARTiが初期化されるため、スタートコンディション検出時に割り込みは必要ありません(「3.3.2 STACビットの使い方(UARTi初期化機能)」参照)。

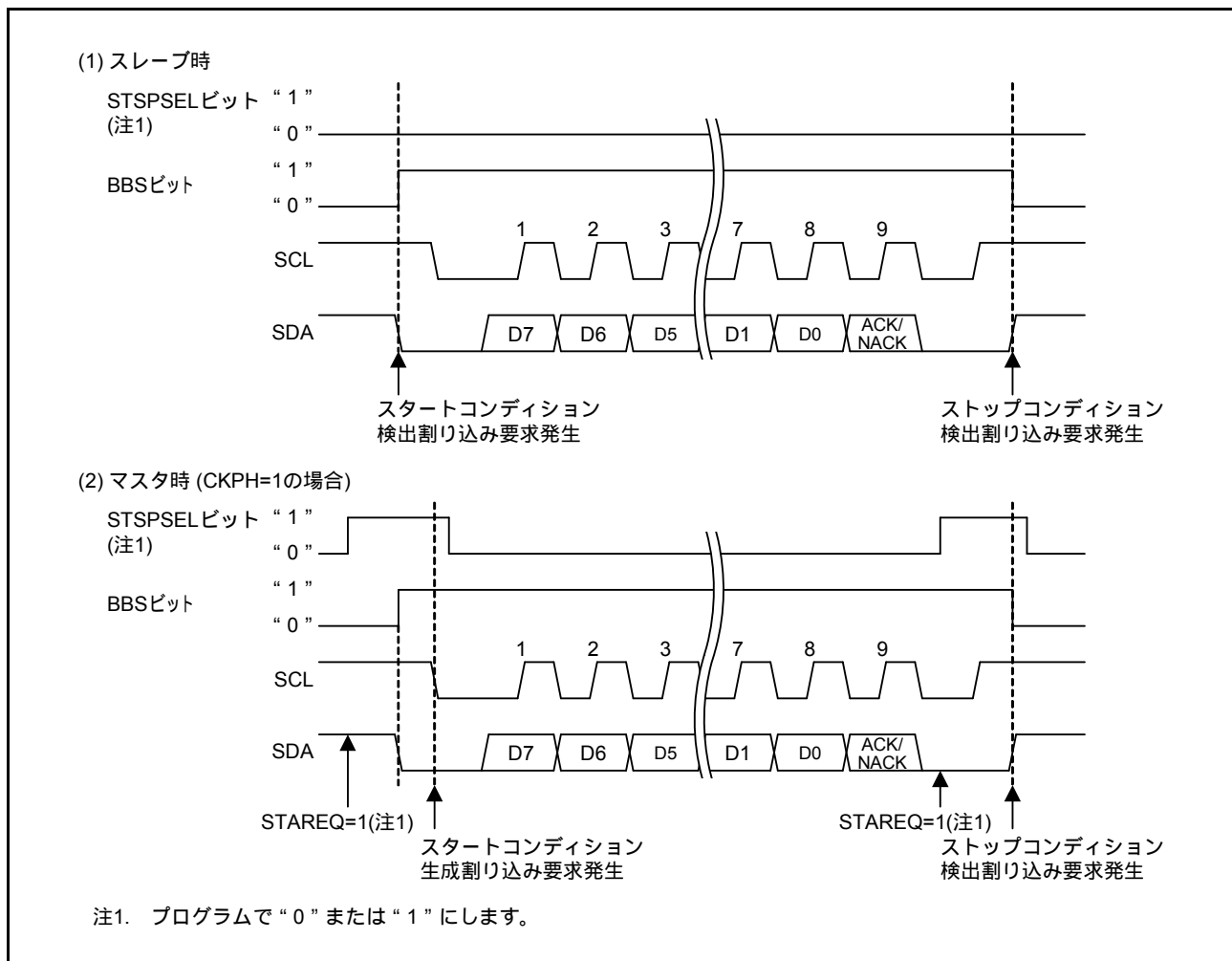


図 7.1 スタートコンディション/ストップコンディション割り込み

7.2 受信 / 送信割り込み

図 7.2 に受信 / 送信割り込みのタイミングを示します。

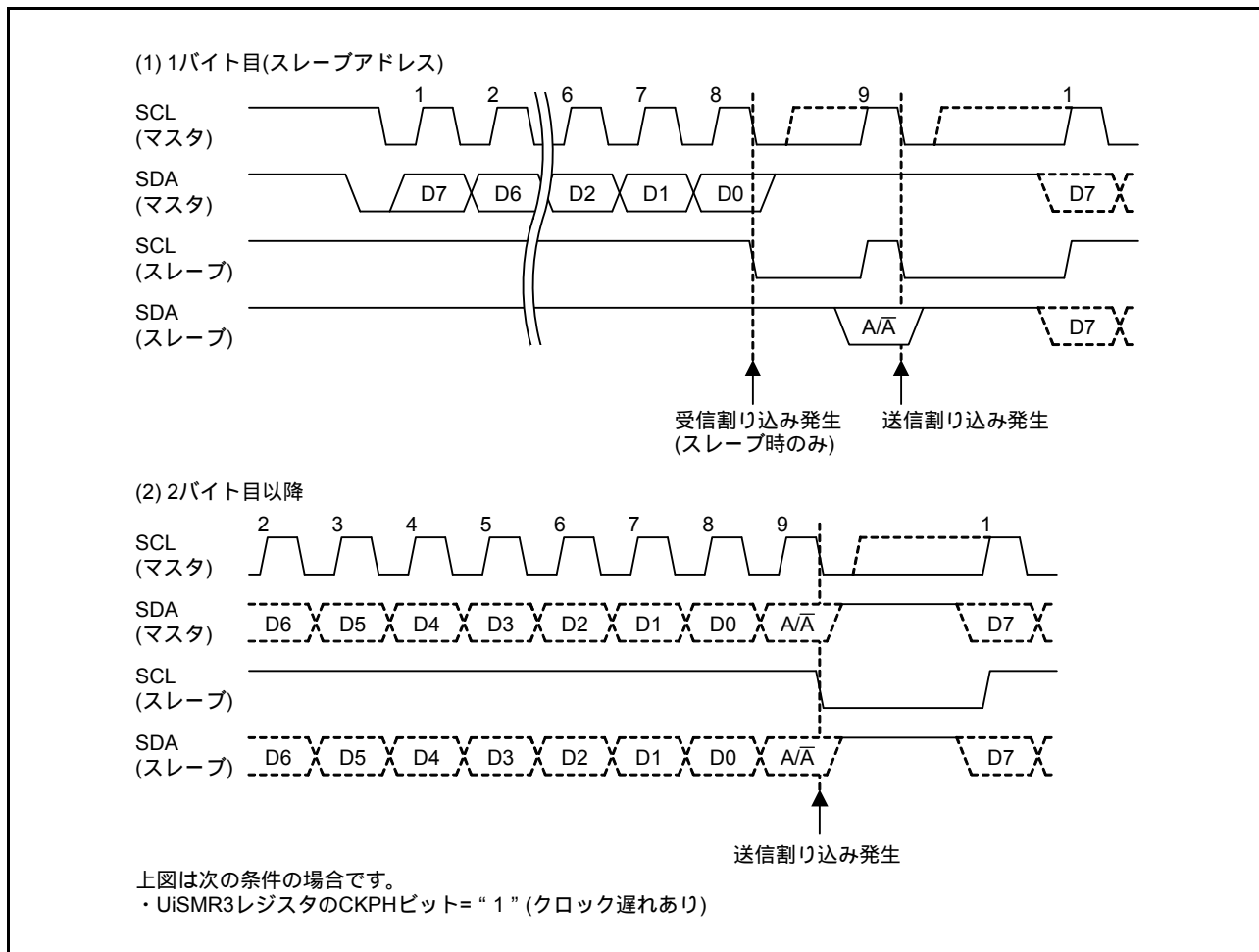


図 7.2 受信 / 送信割り込みのタイミング

7.2.1 受信割り込み

受信割り込みは、SCLクロックの8ビット目の立ち下がりで発生します。

UiSMR3レジスタのCKPHビットが“1”(クロック遅れあり)の場合、受信割り込み処理でUiRBレジスタを読み出すと、受信データはビットの位置が変化しています(「図 4.3 クロック遅延機能」参照)。

以下に、受信割り込み時の処理を示します。
?ユーザの仕様に応じて、処理を追加してください。

スレーブ送信 / 受信

(1) 1バイト目(スレーブアドレス)

- スレーブアドレス確認
 - <スレーブアドレス一致時>
 - ACK生成(ACKD=0、ACKC=1)
 - 9ビット受信後のSCL“L”固定を設定(SWC9=1)、8ビット受信後のSCL“L”固定を開放(SWC=0)
 - <スレーブアドレス不一致時>
 - NACK生成(ACKD=1、ACKC=1)
 - 受信 / 送信割り込みを禁止
 - スタートコンディション / ストップコンディション割り込みを許可
 - 9ビット受信後のSCL“L”固定を禁止(SWC9=0)、8ビット受信後のSCL“L”固定を開放(SWC=0)

7.2.2 送信割り込み

I²Cモード時は、UiC1レジスタのUiIRSビットを“1”(UARTi送信割り込み要因は送信完了(TXEPT=1))にして使用してください。

UiSMR3レジスタのCKPHビットが“1”(クロック遅れあり)のとき、送信割り込みは、SCLクロックの9ビット目の立ち下がりで発生します。

以下に、送信割り込み時の処理を示します。
?ユーザの仕様に応じて、処理を追加してください。

マスタ送信

(1) 1バイト目(スレーブアドレス)

- ACK/NACK確認(スレーブアドレス一致確認)
- 2バイト目の送信データ設定(ACK時)
- ストップコンディションまたはリスタートコンディション生成(NACK時)

(2) 2バイト目以降

- ACK/NACK確認
- 次バイトの送信データ設定(ACK時)
- ストップコンディションまたはリスタートコンディション生成(NACK時またはリスタート時または最終バイト時)

マスタ受信

- (1) 1バイト目(スレーブアドレス)
 - ACK/NACK確認(スレーブアドレス一致確認)
 - 2バイト目のACKデータ設定(ACK時)(次回受信の準備)
 - ストップコンディションまたはリスタートコンディション生成(NACK時)
- (2) 2バイト目以降
 - 次バイトのACKデータ設定(次回受信の準備)
- (3) (最終-1)バイト目
 - 次バイトのNACKデータ設定(次回受信の準備)
- (4) 最終バイト目
 - ストップコンディションまたはリスタートコンディション生成

スレーブ送信

- (1) 1バイト目(スレーブアドレス)
 - 受信割り込み処理で設定したACKデータ出力を禁止(ACKC=0)
 - 2バイト目の送信データ設定
 - SCL“L”固定を開放(SWC9=0)、設定直後に、2バイト目の9ビット受信後のSCL“L”固定を設定(SWC9=1)。
- (2) 2バイト目以降
 - ACK/NACK確認
 - 次バイトの送信データ設定(ACK時)
 - SCL“L”固定を開放(SWC9=0)、設定直後に、次バイトの9ビット受信後のSCL“L”固定を設定(SWC9=1)。

スレーブ受信

- (1) 1バイト目(スレーブアドレス)
 - 受信割り込み処理で設定したACKデータ出力を禁止(ACKC=0)
 - 2バイト目のACKデータ設定(次回受信の準備)
 - SCL“L”固定を開放(SWC9=0)、設定直後に、2バイト目の9ビット受信後のSCL“L”固定を設定(SWC9=1)。
- (2) 2バイト目以降
 - 次バイトのACKデータ設定(次回受信の準備)
 - SCL“L”固定を開放(SWC9=0)、設定直後に、次バイトの9ビット受信後のSCL“L”固定を設定(SWC9=1)。

8. UARTi特殊モード1(I²Cモード)使用時の注意事項

8.1 スレーブ送信/スレーブ受信時のUiTBレジスタのデータ設定手順

スレーブ送信/スレーブ受信時は、下記の手順に従いUiTBレジスタにデータを書いてください。

- 1バイト目のデータ(スレーブアドレス)受信で
 - (1) 受信割り込みで、UiTBレジスタに2バイト目のデータを書いてください。
 - (2) 送信割り込みで、UiTBレジスタに3バイト目のデータを書いてください。
- 2バイト目以降のデータ受信で
送信割り込みごとに、UiTBレジスタに4バイト目以降のデータを順番に書いてください。

8.2 電気的特性

M16Cファミリ、R8Cファミリの電気的特性は、I²Cバス規格と一部異なる点があります。

8.2.1 “L”レベル/“H”レベル入力電圧、“L”レベル出力電圧

“L”入力電圧、“H”入力電圧、および“L”出力電圧は、I²Cバス規格と異なる場合があります。電気的特性はマイコンにより異なりますので、ハードウェアマニュアルで確認してください。

< M16C/62Pグループの場合 >

2.7V ~ 5.5V動作時

“H”入力電圧(V_{IH}) = 最小0.8 V_{CC} (保証値)

“L”入力電圧(V_{IL}) = 最大0.2 V_{CC} (保証値)

$V_{CC} = 5V$ 、 $I_{OL} = 5mA$ 時

“L”出力電圧(V_{OL}) = 最大2.0 V (保証値)

< I²Cバス規格 >

“H”入力電圧(V_{IH}) = 最小0.7 V_{CC}

“L”入力電圧(V_{IL}) = 最大0.3 V_{CC}

$I_{OL} = 6mA$ 時

“L”出力電圧(V_{OL}) = 最大0.6 V

8.2.2 コンディション検出時のセットアップ時間およびホールド時間

スレープ時、スタートコンディション/ストップコンディション検出時のセットアップ時間およびホールド時間は、I²Cバス規格と異なる場合があります。

I²Cバス規格のFast-Modeにおいて、スタートコンディション/ストップコンディションのセットアップ時間、ホールド時間はどちらも最小 600ns です (Standard-Mode は最小 4.0μs)。それに対して、M16Cファミリのセットアップ時間、ホールド時間は、UiBRGカウントソースの最小6サイクルです (マイコンによってサイクル数は変わります)。

Fast-Modeの場合、UiBRGカウントソースを 10MHz で使用するときには最小 600ns となり、I²Cバス規格を満たしますが、10MHz未滿で使用するときには、I²Cバス規格を満たすことができなくなります。

Standard-Modeの場合、UiBRGカウントソースを 1.5MHz で使用するときには最小 4.0μs となり、I²Cバス規格を満たしますが、1.5MHz未滿で使用するときには、I²Cバス規格を満たすことができなくなります。

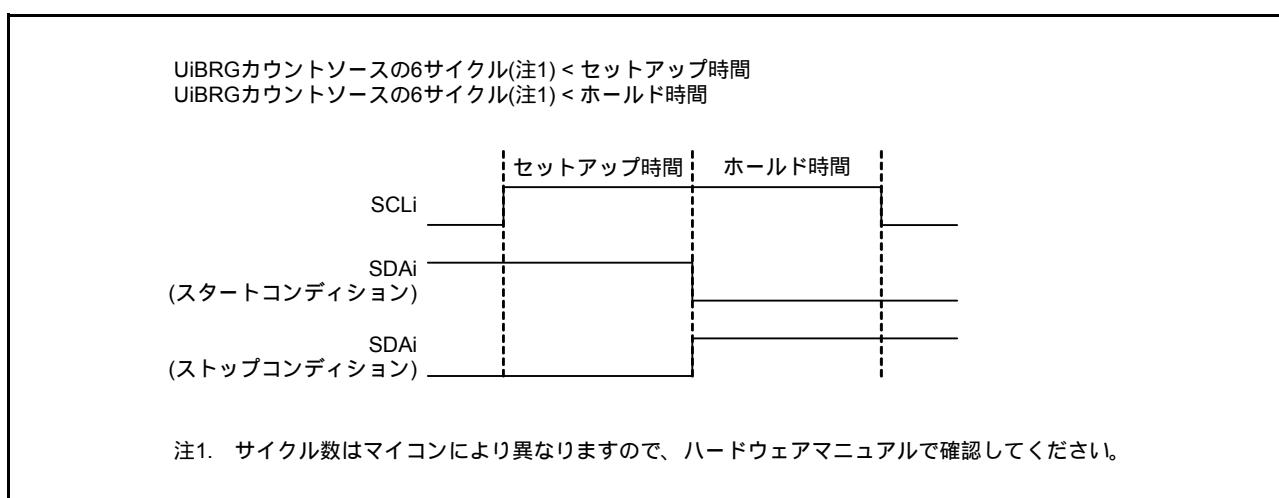


図 8.1 スタートコンディション/ストップコンディション検出時のセットアップ時間、ホールド時間

8.2.3 コンディション生成時のセットアップ時間およびホールド時間

スタートコンディション生成時、SCLクロックの半サイクル分の時間がホールド時間 ($t_{HD:STA}$) となります。ストップコンディション生成時、SCLクロックの半サイクル分の時間がセットアップ時間 ($t_{SU:STO}$) となります。

また、SDAデジタル遅延機能を使用する場合、遅延時間を考慮する必要があります(「3.3.3 DL2 ~ DL0ビットの使い方(SDAデジタル遅延機能)」参照)。

以下にコンディション生成時のホールド時間およびセットアップ時間の算出例を示します。

< 100kbps設定時の算出例 >

- ・ UiBRG カウントソース : $f_1 = 20\text{MHz}$
- ・ UiBRG レジスタの設定値 : $n = 100 - 1$
- ・ SDA デジタル遅延設定値 : $DL2 \sim DL0 = 101b$ (UiBRG カウントソースの5 ~ 6サイクル) の場合、

$$\begin{aligned}
 f_{SCL}(\text{理論}) &= f_1 / (2(n+1)) &= 20\text{MHz} / (2 \times (99+1)) &= 100\text{kbps} \\
 t_{DL} &= \text{遅延サイクル数} / f_1 &= 6 / 20\text{MHz} &= 0.3\mu\text{s} \\
 t_{HD:STA}(\text{理論}) &= 1 / (2f_{SCL}(\text{理論})) &= 1 / (2 \times 100\text{kbps}) &= 5\mu\text{s} \\
 t_{SU:STO}(\text{理論}) &= 1 / (2f_{SCL}(\text{理論})) &= 1 / (2 \times 100\text{kbps}) &= 5\mu\text{s} \\
 t_{HD:STA}(\text{実効}) &= t_{HD:STA}(\text{理論}) - t_{DL} &= 5\mu\text{s} - 0.3\mu\text{s} &= 4.7\mu\text{s} \\
 t_{SU:STO}(\text{実効}) &= t_{SU:STO}(\text{理論}) + t_{DL} &= 5\mu\text{s} + 0.3\mu\text{s} &= 5.3\mu\text{s}
 \end{aligned}$$

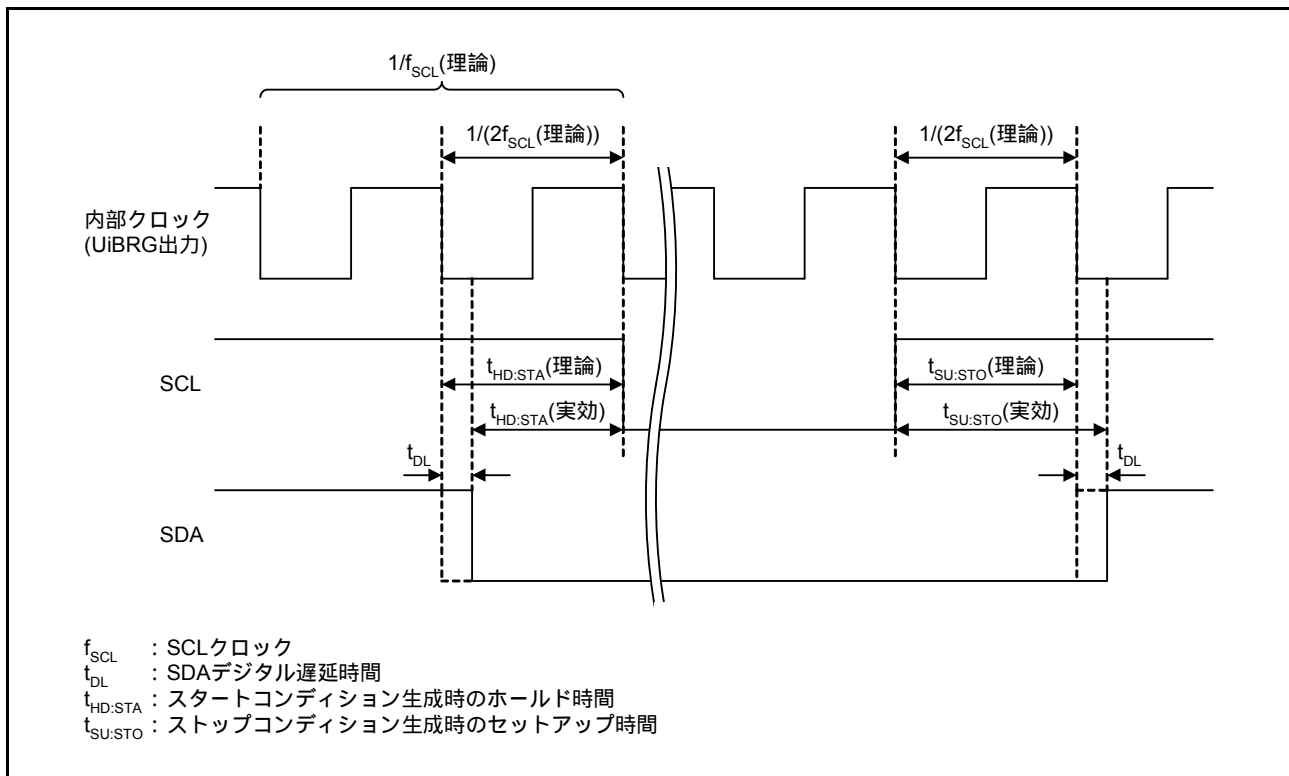


図 8.2 スタートコンディション/ストップコンディション生成時のホールド時間、セットアップ時間

8.3 UiBRGカウントソースによる最大転送速度の制限

SCLクロックのレベルを認識するまでの時間はサンプリング周期に依存し、最大UiBRGカウントソースの3クロック分を要します。したがって、動作周波数およびUiC0レジスタのCLK1～CLK0ビットで選択したUiBRGカウントソースの速度によって、M16Cファミリ、R8Cファミリに接続可能なI²Cバスの最大転送速度が制限されます。以下の条件を満たす転送速度で使用しない場合、ビットずれを起こす可能性があります。

$$I^2C \text{バスの最大転送速度(Hz)} < \text{UiBRGカウントソース(Hz)} / 3$$

(例1) 源発振10MHz、UiBRGカウントソースにf32を選択した場合
ビットずれを起こさない最大転送速度(Hz) < (10MHz/32) / 3 = 104kbps
この場合のI²Cバスの最大転送速度は、104kbpsになります。

(例2) 源発振10MHz、UiBRGカウントソースにf8を選択した場合
ビットずれを起こさない最大転送速度(Hz) < (10MHz/8) / 3 = 416kbps
この場合のI²Cバスの最大転送速度は、400kbps(Fast-Modeの最大値)になります。

8.4 使用機能の制限

8.4.1 UiSMR2レジスタのSWC2ビット(SCLウェイト機能)

SWC2ビットは、通常、使用する必要はありません。

SWC2ビットを“1”(“L”出力)にすると、送受信中でもSCLi端子を“L”に固定できます。SWC2ビットを“0”(転送クロック)にすると、SCLi端子の“L”固定は解除され、SCLクロックが出力されます。

8.4.2 UiSMR2レジスタのSDHIビット(SDA出力禁止機能)

SDHIビットは、通常、使用する必要はありません。

スレープ時、スレープアドレス不一致のときはSDAi端子からの出力を禁止にします。その場合、UiTBレジスタに“01FFh”を設定するか、SDA出力禁止機能を使用すると、SDAi端子をハイインピーダンス状態にできます。

SDA出力禁止機能は、SDAi端子をハイインピーダンス状態にする機能で、SDHIビットを“1”(禁止)にすると有効になります。SDHIビットを“0”(許可)にすると、次のSCLクロックに同期して、UiTBレジスタに設定した値がSDAi端子から出力されます。

8.4.3 スレープ時のリスタートコンディション

I²Cモードではスレープ時のリスタートコンディションの検出は対応していません。

9. 参考ドキュメント

M16Cファミリ、R8Cファミリのユーザーズマニュアル ハードウェア編
(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート / テクニカルニュース
(最新の情報をルネサス エレクトロニクスホームページから入手してください。)

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ
<http://japan.renesas.com/>

お問合せ先
<http://japan.renesas.com/inquiry>

改訂記録	M16Cファミリ、R8Cファミリ UARTi特殊モード1を使用したI ² Cバスインタフェース
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.01.25	–	初版発行
1.01	2010.09.01	–	R8Cファミリ追加
		–	全体的に記述修正
		3	3.2.3 SCLクロックの数え方追加、図3.2追加
		14	6.2 8ビット目(b8)をb8に変更
		19	スレーブ送信/スレーブ受信時のUiTBレジスタのデータ設定手順を8.1として追加
1.02	2010.12.20	3	3.1章改訂

すべての商標および登録商標は、それぞれの所有者に帰属します。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。

外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレス（予約領域）のアクセス禁止

【注意】リザーブアドレス（予約領域）のアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違うと、内部 ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

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

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

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



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/inquiry>