

LC717A00



ON Semiconductor®

<http://onsemi.jp>

CMOS LSI

静電容量タッチセンサ用
容量デジタルコンバータLSI

アプリケーションノート

概要

LC717A00 のレジスタや各種機能について説明します。
また、その制御方法の例を示します。

目次

第 1 章 レジスタ

- 1.1 レジスタマップ・・・・・・・・・・・・・・・・・・1-1
- 1.2 レジスタ詳細・・・・・・・・・・・・・・・・・・1-3
- 1.3 推奨設定値・・・・・・・・・・・・・・・・・・1-15

第 2 章 機能

- 2.1 機能説明・・・・・・・・・・・・・・・・・・2-1
 - 2.1.1 インターバルモードとスリープモード・・・・・・・・2-1
 - 2.1.2 ショートインターバルモードとロングインターバルモード・・・・・・・・2-1
 - 2.1.3 キャリブレーション・・・・・・・・・・・・・・・・2-2
 - 2.1.4 ノイズ除去・・・・・・・・・・・・・・・・2-2
 - 2.1.5 タッチ ON 自動キャンセル機能・・・・・・・・2-2
- 2.2 計測機能・・・・・・・・・・・・・・・・・・2-3
 - 2.2.1 計測タイミングとロングインターバル/ショートインターバルの設定との関係・・2-3
 - 2.2.2 動的オフセットキャリブレーションの説明・・・・・・・・2-10
 - 2.2.3 計測間隔の概算式・・・・・・・・・・・・・・・・2-12
 - 2.2.4 タッチ ON 自動キャンセル機能の説明・・・・・・・・2-12
 - 2.2.5 タッチ OFF 判定時の動的しきい値制御の説明・・・・・・・・2-13

第 3 章 動作シーケンス

- 3.1 外部リセット時の動作シーケンス例・・・・・・・・3-1
- 3.2 インターバルモード時の動作シーケンス例・・・・・・・・3-3
- 3.3 スリープモード時の動作シーケンス例・・・・・・・・3-6

第 4 章 制御マイコンによる制御例

- 4.1 制御フロー例・・・・・・・・・・・・・・・・4-1
 - 4.1.1 インターバルモードに設定し、INTOUT を使って計測結果をリードする場合・・・・4-1
 - 4.1.2 インターバルモードに設定し、INTOUT を使わずに計測結果をリードする場合・・4-4
 - 4.1.3 スリープモードに設定し、INTOUT を使って計測結果をリードする場合・・・・4-7
 - 4.1.4 インターバルモードに設定し、Pout0~7 の出力をリードする場合・・・・4-10
 - 4.2 サンプルソース（擬似コード形式）・・・・4-13
 - 4.2.1 インターバルモードに設定し、INTOUT を使って計測結果をリードする場合・・・・4-16
 - 4.2.2 インターバルモードに設定し、INTOUT を使わずに計測結果をリードする場合・・4-20
 - 4.2.3 スリープモードに設定し、INTOUT を使って計測結果をリードする場合・・・・4-24
-

LC717A00 アプリケーションノート

1. レジスタ

1.1 レジスタマップ

Register Address	Reset Value	名称	説明
0x00	0xFF	Use Channel Register	使用チャンネル設定 Cin0～7
0x01	0x60/0xD0	Cin0 Gain Register	Cin0 ゲイン設定
0x02	0x60/0xD0	Cin1 Gain Register	Cin1 ゲイン設定
0x03	0x60/0xD0	Cin2 Gain Register	Cin2 ゲイン設定
0x04	0x60/0xD0	Cin3 Gain Register	Cin3 ゲイン設定
0x05	0x60/0xD0	Cin4 Gain Register	Cin4 ゲイン設定
0x06	0x60/0xD0	Cin5 Gain Register	Cin5 ゲイン設定
0x07	0x60/0xD0	Cin6 Gain Register	Cin6 ゲイン設定
0x08	0x60/0xD0	Cin7 Gain Register	Cin7 ゲイン設定
0x09	0x32	Cin0 Threshold Register	Cin0 しきい値設定
0x0A	0x32	Cin1 Threshold Register	Cin1 しきい値設定
0x0B	0x32	Cin2 Threshold Register	Cin2 しきい値設定
0x0C	0x32	Cin3 Threshold Register	Cin3 しきい値設定
0x0D	0x32	Cin4 Threshold Register	Cin4 しきい値設定
0x0E	0x32	Cin5 Threshold Register	Cin5 しきい値設定
0x0F	0x32	Cin6 Threshold Register	Cin6 しきい値設定
0x10	0x32	Cin7 Threshold Register	Cin7 しきい値設定
0x11	0x00	Cin0 Data Register	Cin0 計測値
0x12	0x00	Cin1 Data Register	Cin1 計測値
0x13	0x00	Cin2 Data Register	Cin2 計測値
0x14	0x00	Cin3 Data Register	Cin3 計測値
0x15	0x00	Cin4 Data Register	Cin4 計測値
0x16	0x00	Cin5 Data Register	Cin5 計測値
0x17	0x00	Cin6 Data Register	Cin6 計測値
0x18	0x00	Cin7 Data Register	Cin7 計測値
0x19	0x00	Result Data Register	Cin0～7 の ON/OFF 判定結果
0x1A	0x00	Error Status Register	エラー発生状態通知
0x1B	0x00	Error Channel Status Register	各チャンネルのキャリブレーション エラー発生の有無
0x1C	0x0B	Control 1 Register	コントロール 1
0x1D	0x40	Average Count Register	計測データ平均回数
0x1E	0x04	Filter Parameter Register	ノイズ対策用パラメータ
0x1F	0x02	Debounce Count Register	デバウンスカウント

次ページへ続く。

LC717A00 アプリケーションノート

前ページより続く。

Register Address	Reset Value	名称	説明
0x20	0x05	Short Interval Time Register	ショートインターバル時間
0x21	0x01	Long Interval Time Register	ロングインターバル時間
0x22	0x07	Short Interval Dynamic OffCal Cycle Register	ショートインターバル時の動的オフセットキャリブレーション判定実施サイクル
0x23	0x03	Dynamic OffCal Count Plus Register	動的オフセットキャリブレーション実施カウント数(正值側)
0x24	0x03	Dynamic OffCal Count Minus Register	動的オフセットキャリブレーション実施カウント数(負値側)
0x25	0x58	Touch ON Cancel Count Lower Register	タッチ ON キャンセル時間カウント(下位側)
0x26	0x02	Touch ON Cancel Count Higher Register	タッチ ON キャンセル時間カウント(上位側)
0x27 ～ 0x3C	-	-	システム予約領域(W 禁止)
0x3D	0x80	Static OffCal CDAC Base Register	静的オフセットキャリブレーション基準容量値
0x3E	0x00	Measurement Mode Register	計測モード設定
0x3F	-	-	システム予約領域(W 禁止)
0x40	0x00	Control 2 Register	コントロール 2
0x41 ～ 0x6B	-	-	システム予約領域(W 禁止)
0x6C	-	CDAC Offset Plus Register	CDAC オフセット(正側)
0x6D	-	CDAC Offset Minus Register	CDAC オフセット(負側)
0x6E ～ 0x7E	-	-	システム予約領域(W 禁止)
0x7F	0x16/0x17	SLAVE Address Register	I ² C スレーブアドレス
0x80 ～ 0xFF	-	-	システム予約領域(W 禁止)

※ システム予約領域のレジスタの値を読み出しても、その値は保証されません。

LC717A00 アプリケーションノート

1.2 レジスタ詳細

●Use Channel Register

Bit	7	6	5	4	3	2	1	0
Name	Cin7EN	Cin6EN	Cin5EN	Cin4EN	Cin3EN	Cin2EN	Cin1EN	Cin0EN
Reset	1	1	1	1	1	1	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

使用するチャンネルを設定する 8bit のレジスタです。Cin0 が Bit0、Cin7 が Bit7 に対応しています。

CinxEN

0 : 無効チャンネル

1 : 使用チャンネル

●CinX Gain Register (X=0~7)

Bit	7	6	5	4	3	2	1	0
Name	GainX_S3	GainX_S2	GainX_S1	GainX_S0	GainX_F3	GainX_F2	GainX_F1	GainX_F0
Reset	0	1	1	0	0	0	0	0
GAIN=L								
Reset	1	1	0	1	0	0	0	0
GAIN=H								
R/W	RW	RW	RW	RW	RW	RW	RW	RW

LSI 内部には容量変化を検出しアナログ振幅値を出力する 2 段のアンプがあります（初段アンプと次段アンプ）。このレジスタは、チャンネル毎にアンプのゲインを設定する 8bit のレジスタです。Cin0 のゲインは Cin0 Gain Register、Cin7 のゲインは Cin7 Gain Register にて設定します。基本的には、初段アンプのゲインは最小で使用し、通常は次段アンプのゲインのみで調整を行ってください。

GainX_F0~3 : 下位 4bit により初段アンプのゲインを設定します。(最小 : 0000b~最大 : 1111b)

GainX_S0~3 : 上位 4bit により次段アンプのゲインを設定します。(最小 : 0000b~最大 : 1111b)

初期値は LSI 内部初期化時の GAIN 端子への入力が “Low” の場合は 0x60、“High” の場合は 0xD0 です。

<計算式>

初段アンプ出力値 : $\Delta V_1 = (\Delta C / C_f) \times V_{CDRV}$ $\Delta V_1 < 0.8 \times V_{DD}$

次段アンプ出力値 : $\Delta V_2 = \Delta V_1 \times ga_2$ $\Delta V_2 < 0.8 \times V_{DD}$

ΔC : 入力容量変化量(タッチ時の容量変化分)

C_f : 初段アンプのゲイン設定値 (LSI 内部のフィードバックコンデンサ容量)

V_{CDRV} : Cdrv 端子の High 出力電圧 (= V_{DD})

ga_2 : 次段アンプのゲイン設定値(倍率)

次段アンプのゲイン設定				
GainX_S3	GainX_S2	GainX_S1	GainX_S0	ゲイン[倍]
0	0	0	0	1(最小)
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7
0	1	1	1	8
1	0	0	0	9
1	0	0	1	10
1	0	1	0	11
1	0	1	1	12
1	1	0	0	13
1	1	0	1	14
1	1	1	0	15
1	1	1	1	16(最大)

初段アンプのゲイン設定				
GainX_F3	GainX_F2	GainX_F1	GainX_F0	Cf [fF]
0	0	0	0	1600(最小)
0	0	0	1	1500
0	0	1	0	1400
0	0	1	1	1300
0	1	0	0	1200
0	1	0	1	1100
0	1	1	0	1000
0	1	1	1	900
1	0	0	0	800
1	0	0	1	700
1	0	1	0	600
1	0	1	1	500
1	1	0	0	400
1	1	0	1	300
1	1	1	0	200
1	1	1	1	100(最大)

LC717A00 アプリケーションノート

●CinX Threshold Register (X=0~7)

Address	Bit	7	6	5	4	3	2	1	0
0x09 ~0x10	Name	CinXTH7	CinXTH6	CinXTH5	CinXTH4	CinXTH3	CinXTH2	CinXTH1	CinXTH0
	Reset	0	0	1	1	0	0	1	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

各チャネルの計測データから ON/OFF を判定するためのしきい値を 0~127 まですで設定する 8bit のレジスタです。Cin0 のしきい値は Cin0 Threshold Register にて設定し、Cin7 のしきい値は Cin7 Threshold Register にて設定します。

ゲインの変更等によりタッチ ON 判定のしきい値を変更したい場合に使用します。

(タッチ OFF 判定は、Measurement Mode Register 【Address=0x3E】の TOFFTH ビットの説明を参照して下さい。)

正の値のみ設定可能 (CinXTH7 は “0” 固定。 “1” は禁止) です。しきい値の初期値は 50 (0x32) です。

CinXTH7	CinXTH6	CinXTH5	CinXTH4	CinXTH3	CinXTH2	CinXTH1	CinXTH0	しきい値
0	0	0	0	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	1	1	1	1	1	1	1	127
1	0	0	0	0	0	0	0	禁止
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	禁止
1	1	1	1	1	1	1	1	禁止

●CinX Data Register (X=0~7)

Address	Bit	7	6	5	4	3	2	1	0
0x11 ~0x18	Name	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
	Reset	0	0	0	0	0	0	0	0
	R/W	R	R	R	R	R	R	R	R

各チャネルを計測して得られた計測データ (-128~0~127) を読み出すためのレジスタです。格納される値は 2 の補数表現です (0x80~0x00~0x7F)。Cin0 の計測データは Cin0 Data Register に格納し、Cin7 の計測データは Cin7 Data Register に格納します。

制御マイコン側でタッチ状況を判定する際に、この計測データを用います。

●Result Data Register

Address	Bit	7	6	5	4	3	2	1	0
0x19	Name	Cin7ACT	Cin6ACT	Cin5ACT	Cin4ACT	Cin3ACT	Cin2ACT	Cin1ACT	Cin0ACT
	Reset	0	0	0	0	0	0	0	0
	R/W	R	R	R	R	R	R	R	R

各チャネルのタッチ ON/OFF 状況を通知するための 8bit のレジスタです。Cin0 が Bit0、Cin7 が Bit7 に対応します。

CinxACT

0: タッチ OFF 判定 (初期値)

1: タッチ ON 判定

LC717A00 アプリケーションノート

●Error Status Register

Address	Bit	7	6	5	4	3	2	1	0
0x1A	Name	SYSERR	–	–	–	–	–	–	CALERR
	Reset	0	0	0	0	0	0	0	0
	R/W	R	R	R	R	R	R	R	R

エラー発生の有無を通知するためのレジスタです。

SYSERR

- 0 : エラー無し(初期値)
- 1 : システムエラー発生

CALERR

- 0 : エラー無し(初期値)
- 1 : キャリブレーションエラー発生

“1” にセットされた SYSERR ビットを “0” に戻すためには Control 2 Register 【Address=0x40】 の SoftRst ビットを “1” にセットして LSI をソフトリセットするか、またはパワーオンリセットをしてください。

“1” にセットされた CALERR ビットを “0” に戻すためには Control 1 Register 【Address=0x1C】 の ParaCh ビットを “1”、WriteReq ビットを “1” にセットしてパラメータ更新処理を実施してください。

●Error Channel Status Register

Address	Bit	7	6	5	4	3	2	1	0
0x1B	Name	Cin7ERR	Cin6ERR	Cin5ERR	Cin4ERR	Cin3ERR	Cin2ERR	Cin1ERR	Cin0ERR
	Reset	0	0	0	0	0	0	0	0
	R/W	R	R	R	R	R	R	R	R

エラー発生時、Cin0～Cin7 のキャリブレーションエラーコード、システムエラーコードを通知するレジスタです。Cin0 が Bit0、Cin7 が Bit7 に対応します。

CinxERR

- 0 : キャリブレーションエラー発生なしチャンネル(初期値)
- 1 : キャリブレーションエラー発生ありチャンネル
- ※ システムエラーが発生した場合、全てのビットが “0” になります。

すべてのビットを “0” に戻すためには、Control 1 Register 【Address=0x1C】 の ParaCh ビットを “1”、WriteReq ビットを “1” にセットしてパラメータ更新処理を実施してください。

LC717A00 アプリケーションノート

●Control 1 Register

Address	Bit	7	6	5	4	3	2	1	0
	Name	WriteReq	Rsvd6	Rsvd5	Rsvd4	IntMode	ParaCh	StaCal	Measure
0x1C	Reset	0	0	0	0	1	0	1	1
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

WriteReq

各制御ビット(IntMode, ParaCh, StaCal, Measure)の設定反映フラグ。

これらの制御ビットの設定は WriteReq ビットに“1”がセットされた場合に初めて LSI の動作に反映されます。“0”がセットされた状態では反映されません。インターバル状態もしくはスリープ状態から抜けたタイミングで設定反映処理が実行されます。設定が反映されると WriteReq ビットは自動的に“0”になります。

0：設定を反映しない(初期値)

1：設定を反映する

Rsvd6～Rsvd4 “0” 固定。

IntMode

インターバルモードフラグ。各モードについては 2.1 機能説明の項を参照してください。

0：スリープモードで動作

1：インターバルモードで動作(初期値)

ParaCh

レジスタ設定変更要求フラグ。

パラメータ更新要求ありの場合、本レジスタ以外の各レジスタ設定の内容を反映するパラメータ変更処理を実行します。パラメータ変更処理後、ParaCh ビットは自動的に“0”になります。

パラメータ変更処理完了後、Result Data Register 【Address=0x19】、Error Status Register 【Address=0x1A】、Error Channel Status Register 【Address=0x1B】の内容をクリアし、INTOUT をネグートします。くわえて、動的オフセットキャリブレーション処理用内部カウンタ、デバウンス処理用内部カウンタ、タッチ ON 自動キャンセル処理用内部カウンタの値を 0 クリアします。

0：パラメータ変更要求なし(初期値)

1：パラメータ変更要求あり

StaCal

静的オフセットキャリブレーション要求フラグ。要求ありの場合、静的オフセットキャリブレーションを実行します。実行後に StaCal ビットは自動的に“0”になります。

静的オフセットキャリブレーションは、各入力チャネル(Cin0～7)の寄生容量に対して容量 D/A コンバータのオフセット調整を行い、それぞれのチャネルに対応した最適なオフセット値(正值)とオフセット値(負値)を決定します。

0：静的オフセットキャリブレーション要求なし

1：静的オフセットキャリブレーション要求あり(初期値)

※LC717A00が静的オフセットキャリブレーション処理を実行している最中に制御マイコンが WriteReq ビットを1、StaCal ビットを0にして Control 1 Register にライトしても、LC717A00の静的オフセットキャリブレーション処理がすぐに中断しないことに注意してください。LC717A00で現在行われている静的オフセットキャリブレーション処理は中断することなく最後まで実行されます。

次ページへ続く。

LC717A00 アプリケーションノート

前ページより続く。

Measure

計測フラグ。通常は“1” 固定。

0：計測を行わない

1：計測を行う(初期値)

※ 基本的に WriteReq、ParaCh、StaCal の全ビットが“0” の場合にのみ、制御マイコンから本レジスタへの書き込みを行って下さい。

例外として、LSI の初期化完了 (INTOUT アサート) から自動実行される静的オフセットキャリブレーションが終了するまでの間のみ、StaCal ビットが”1” であっても、本レジスタへ 0x80 または 0x88 の値を書き込むことがあります。詳細は 4. 制御フローを参照して下さい。

※ 処理順番 (WriteReq、ParaCh、StaCal、Measure の全ビットが“1” の場合の処理順番)

- ① 制御ビットの設定反映処理 (WriteReq)。
処理完了後、自動的に WriteReq ビットクリア。
- ② パラメータ更新処理 (ParaCh)。
処理完了後、自動的に ParaCh ビットクリア。
- ③ 静的オフセットキャリブレーション処理 (StaCal)。
処理完了後、自動的に StaCal ビットクリア。
- ④ 計測処理 (Measure)。計測処理は繰り返し。

●Average Count Register

Address 0x1D	Bit	7	6	5	4	3	2	1	0
	Name	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
	Reset	0	1	0	0	0	0	0	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

計測データ平均回数。8、16、32、64、128 回のいずれかを必ず設定してください。
初期値は 64 回 (0x40) です。

AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0	計測データ平均回数
0	0	0	0	1	0	0	0	8 回
0	0	0	1	0	0	0	0	16 回
0	0	1	0	0	0	0	0	32 回
0	1	0	0	0	0	0	0	64 回
1	0	0	0	0	0	0	0	128 回
上記以外								禁止

●Filter Parameter Register

Address 0x1E	Bit	7	6	5	4	3	2	1	0
	Name	FP2_3	FP2_2	FP2_1	FP2_0	FP1_3	FP1_2	FP1_1	FP1_0
	Reset	0	0	0	0	0	1	0	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

ノイズ対策用内部パラメータ。初期値は 0x04 です。

FP1_[3:0]

フィルタパラメータ 1。

FP2_[3:0]

フィルタパラメータ 2。

LC717A00 アプリケーションノート

●Debounce Count Register

Address 0x1F	Bit	7	6	5	4	3	2	1	0
	Name	DCT7	DCT6	DCT5	DCT4	DCT3	DCT2	DCT1	DCT0
	Reset	0	0	0	0	0	0	1	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

各チャネルのタッチ判定が OFF から ON、もしくは ON から OFF に遷移するときのデバウンスカウンタの回数を設定します。

デバウンスカウンタはチャタリング防止のためのカウンタです。初期値は 3 回(0x02)です。

チャネルが OFF から ON になる条件：

計測データが ON/OFF 判定のしきい値を連続して上回った回数が、デバウンスカウンタの値になったとき、そのチャネルが ON になったと判定します。

チャネルが ON から OFF になる条件：

計測データが ON/OFF 判定のしきい値を連続して下回った回数が、デバウンスカウンタの値になったとき、そのチャネルが OFF になったと判定します。

DCT7	DCT6	DCT5	DCT4	DCT3	DCT2	DCT1	DCT0	デバウンスカウンタ
0	0	0	0	0	0	0	0	1 回
0	0	0	0	0	0	0	1	2 回
0	0	0	0	0	0	1	0	3 回
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	0	255 回
1	1	1	1	1	1	1	1	禁止

●Short Interval Time Register

Address 0x20	Bit	7	6	5	4	3	2	1	0
	Name	SIVAL7	SIVAL6	SIVAL5	SIVAL4	SIVAL3	SIVAL2	SIVAL1	SIVAL0
	Reset	0	0	0	0	0	1	0	1
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

ショートインターバルモードのときの計測終了から次の計測開始までのインターバル時間(ショートインターバル時間)を 0~255ms まで 1ms 単位で設定します。

スリープモード設定時は、ショートインターバル時間は 0ms として動作します。

初期値は 5ms(Typ) (0x05)です。

ショートインターバルモードについては 2.1 機能説明の項を参照してください。

SIVAL7	SIVAL6	SIVAL5	SIVAL4	SIVAL3	SIVAL2	SIVAL1	SIVAL0	ショートインターバル時間
0	0	0	0	0	0	0	0	0ms
0	0	0	0	0	0	0	1	1ms
0	0	0	0	0	0	1	0	2ms
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	255ms

LC717A00 アプリケーションノート

●Long Interval Time Register

Address 0x21	Bit	7	6	5	4	3	2	1	0
	Name	LIVAL7	LIVAL6	LIVAL5	LIVAL4	LIVAL3	LIVAL2	LIVAL1	LIVAL0
	Reset	0	0	0	0	0	0	0	1
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

ロングインターバルモードのときの計測終了から次の計測開始までのインターバル時間（ロングインターバル時間）を 0～355ms まで 1ms 単位で設定します。

ロングインターバル時間は、ロングインターバルベース時間（Measurement Mode Register 【Address=0x3E】の LIVALB ビットの設定により 100ms と 0ms を選択）と本レジスタ値の組み合わせにより指定します。

スリープモード設定時は、ロングインターバル時間は 0ms として動作します。

初期値は $100[\text{ms}] + 1[\text{ms}] (0x01) = 101[\text{ms}] (\text{Typ})$ です。

但し、0x00 を設定した場合に限りロングインターバル時間とショートインターバル時間が両方とも 0ms となります。

ロングインターバルモードについては 2.1 機能説明の項を参照してください。

LIVAL7	LIVAL6	LIVAL5	LIVAL4	LIVAL3	LIVAL2	LIVAL1	LIVAL0	ロングインターバル時間	
								LIVALB=0	LIVALB=1
0	0	0	0	0	0	0	0	インターバルなし	インターバルなし
0	0	0	0	0	0	0	1	101ms	1ms
0	0	0	0	0	0	1	0	102ms	2ms
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	355ms	255ms

●Short Interval Dynamic OffCal Cycle Register

Address 0x22	Bit	7	6	5	4	3	2	1	0
	Name	DCYC7	DCYC6	DCYC5	DCYC4	DCYC3	DCYC2	DCYC1	DCYC0
	Reset	0	0	0	0	0	1	1	1
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

ショートインターバルモードのとき、動的オフセットキャリブレーションを実施するかどうかの判定の実施サイクルを設定します。初期値 (0x07) の場合は 7 計測毎に 1 回判定を実施します。

ロングインターバルモードのときはこの設定にかかわらず 1 計測毎に判定を実施します。

ショートインターバルモードのときとロングインターバルモードのときとでは、インターバル時間が違うため、計測の間隔が異なります。この設定は双方の動的オフセットキャリブレーションの実施間隔をそろえる目的に使用します。

スリープモード時は必ず 0x01 を設定して下さい。

なお、0x00 を設定された場合、他レジスタの設定にかかわらず動的オフセットキャリブレーションを実施しません。

DCYC7	DCYC6	DCYC5	DCYC4	DCYC3	DCYC2	DCYC1	DCYC0	ショートインターバルモードの動的オフセットキャリブレーション判定実施サイクル
0	0	0	0	0	0	0	0	動的オフセットキャリブレーションを実施しない
0	0	0	0	0	0	0	1	1 計測毎に 1 回
0	0	0	0	0	0	1	0	2 計測毎に 1 回
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	255 計測毎に 1 回

LC717A00 アプリケーションノート

●Dynamic OffCal Count Plus Register

Bit	7	6	5	4	3	2	1	0
Name	DCALP7	DCALP6	DCALP5	DCALP4	DCALP3	DCALP2	DCALP1	DCALP0
Reset	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

本 LSI では、“タッチが検出されていない状態(全チャネルが OFF 判定)” のとき、計測レンジ正值側の動的オフセットキャリブレーション実施範囲（4～しきい値）に計測値が連続して入った場合、動的オフセットキャリブレーションを自動で実施します。この動的オフセットキャリブレーション実施判定のための連続回数を設定します。初期値は 0x03 で 24 回（設定値×8 回）です。

“タッチが検出されていない状態” とは、「全チャネルの計測データがしきい値未満であるという判定が Debounce Count Register 【Address=0x1F】 で指定したデバウンスカウント以上連続している状態」のことです。（チャネルの判定が OFF であっても、しきい値を超える計測データが 1 回でもあったなら、もう一度デバウンスカウントの回数分しきい値未満の判定が続くまでは動的オフセットキャリブレーションを行いません。）

なお、0x00 に設定された場合、他レジスタの設定にかかわらず動的オフセットキャリブレーションを実施しません。

DCALP7	DCALP6	DCALP5	DCALP4	DCALP3	DCALP2	DCALP1	DCALP0	連続回数
0	0	0	0	0	0	0	0	動的オフセットキャリブレーションを実施しない
0	0	0	0	0	0	0	1	8 回
0	0	0	0	0	0	1	0	16 回
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	2040 回

●Dynamic OffCal Count Minus Register

Bit	7	6	5	4	3	2	1	0
Name	DCALM7	DCALM6	DCALM5	DCALM4	DCALM3	DCALM2	DCALM1	DCALM0
Reset	0	0	0	0	0	0	1	1
R/W	RW	RW	RW	RW	RW	RW	RW	RW

本 LSI では、計測レンジ負値側の動的オフセットキャリブレーション実施範囲（-128～-4）に計測値が一定回数連続して入った場合、動的オフセットキャリブレーションを自動で実施します。この動的オフセットキャリブレーション実施判定のための連続回数を設定します。初期値は 0x03（3 回）です。

なお、0x00 に設定された場合、他レジスタの設定にかかわらず動的オフセットキャリブレーションを実施しません。

DCALM7	DCALM6	DCALM5	DCALM4	DCALM3	DCALM2	DCALM1	DCALM0	連続回数
0	0	0	0	0	0	0	0	動的オフセットキャリブレーションを実施しない
0	0	0	0	0	0	0	1	1 回
0	0	0	0	0	0	1	0	2 回
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	255 回

LC717A00 アプリケーションノート

●Touch ON Cancel Count Lower Register

Address 0x25	Bit	7	6	5	4	3	2	1	0
	Name	TOCL7	TOCL6	TOCL5	TOCL4	TOCL3	TOCL2	TOCL1	TOCL0
	Reset	0	1	0	1	1	0	0	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

●Touch ON Cancel Count Higher Register

Address 0x26	Bit	7	6	5	4	3	2	1	0
	Name	TOCH7	TOCH6	TOCH5	TOCH4	TOCH3	TOCH2	TOCH1	TOCH0
	Reset	0	0	0	0	0	0	1	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

本 LSI では、タッチ検出状態 (ON 状態) を一定回数以上連続して計測した場合、静的オフセットキャリブレーションを自動で実施しその状態をキャンセルします。このタッチ ON 自動キャンセル処理実施判定のための回数を 16bit で設定します。

Touch ON Cancel Count Lower Register にて下位 8bit、Touch ON Cancel Count Higher Register にて上位 8bit を設定します。

初期値ではタッチ検出状態を連続して 600 回 (0x0258) 計測した場合に静的オフセットキャリブレーションを実施しキャンセルします。

なお、設定を 0x0000 にした場合、タッチ検出状態が長く続いてもキャンセル処理を行いません。

TOCH								TOCL								判定回数
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	キャンセル処理なし
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1 回
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2 回
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	65535 回

●Static OffCal CDAC Base Register

Address 0x3D	Bit	7	6	5	4	3	2	1	0
	Name	DACB7	DACB6	DACB5	DACB4	DACB3	DACB2	DACB1	DACB0
	Reset	1	0	0	0	0	0	0	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

静的オフセットキャリブレーション実施時の基準容量の値を設定します。

静的オフセットキャリブレーションを実施しても計測の基準値 (計測の原点) が計測範囲の中央に合わないときに、この値を変更すると解決する場合があります。設定可能な値は、0x20、0x40、0x80 のみで、それ以外の値は禁止です。初期値は 4pF の 0x80 です。

DACB7	DACB6	DACB5	DACB4	DACB3	DACB2	DACB1	DACB0	静的オフセットキャリブレーション実施時の基準容量値
0	0	1	0	0	0	0	0	1pF
0	1	0	0	0	0	0	0	2pF
1	0	0	0	0	0	0	0	4pF
上記以外								禁止

LC717A00 アプリケーションノート

●Measurement Mode Register

Address 0x3E	Bit	7	6	5	4	3	2	1	0
	Name	Rsvd7	Rsvd6	PDCLP	TOFFTH	LIVALB	Rsvd2	Rsvd1	Rsvd0
	Reset	0	0	0	0	0	0	0	0
	R/W	RW	RW	RW	RW	RW	RW	RW	RW

計測モードを設定するためのレジスタです。Bit3～5 以外は “0” 固定です。

PDCLP

正值側動的オフセットキャリブレーション方式選択フラグ。

- 0：正值側の動的オフセットキャリブレーションは、全使用チャンネルの計測値がしきい値以下の場合のみ、判定/実施します。（初期値）
- 1：正值側の動的オフセットキャリブレーションを個別の使用チャンネル毎に判定/実施します。
※例えば、ボード上のすべてのスイッチ間隔が互いに十分離れており、あるスイッチをタッチしたときに別スイッチの計測データにまったく影響がない場合は、本ビットを 1 に設定することができます。

TOFFTH

タッチ ON からタッチ OFF に遷移する際のしきい値をタッチ ON 時における検出値のピークの 3/4 と 1/2 から選択します。但し CinX Threshold Register 【Address=0x09～0x10】 の値の方が大きい場合は CinX Threshold Register の値を優先します。

- 0：タッチ中の検出値のピークの 3/4 (初期値)
- 1：タッチ中の検出値のピークの 1/2

LIVALB

ロングインターバルベース時間選択フラグ

- 0：ロングインターバルのベース時間は 100[ms] (Typ) （初期値）
- 1：ロングインターバルのベース時間は 0[ms]

Rsvd0～2, 6～7 “0” 固定。

LC717A00 アプリケーションノート

●Control 2 Register

Address	Bit	7	6	5	4	3	2	1	0
	Name	SoftRst	Rsvd6	Rsvd5	ErrOut	Rsvd3	Rsvd2	IntOut	WakeUp
	Reset	0	0	0	0	0	0	0	0
0x40	R/W	RW	RW	RW	RW	RW	RW	RW	RW

SoftRst

ソフトリセットビット。“0” から “1” にセットするとソフトリセットを行います。

ソフトリセットを行うと、3.1 外部リセット時の動作シーケンス例における内部リセット解除直後から動作を開始します。

0 : 通常動作(初期値)

1 : ソフトリセット

ErrOut

ERROR 端子の出力制御ビット。エラーが発生していない通常時は “Low” をセットしており、エラーが発生すると自動で “High” をセットします。

ERROR 端子をエラー通知ピンとして使用するためには、Control 2 Register に対して書き込みを行う際に ErrOut のビット内容を保持する必要があります。もし ErrOut のビット内容が書き換わった場合、ERROR 端子の状態もそれに合わせて変化してしまいます。

0 : ERROR 端子出力レベルを “Low” に設定(初期値)

1 : ERROR 端子出力レベルを “High” に設定

IntOut

INTOUT 信号の制御ビット。INTOUT 信号をネゲートするときには、本ビットに “0” をセットしてください。

なお、INTOUT 信号をネゲートするために制御マイコンから書き込みを行う際は、インターバル期間中またはスリープ期間中に本ビットに “0” をセットして INTOUT 信号をネゲートしてください。

INTOUT 信号アサート時の INTOUT 出力レベルは “High”、ネゲート時は “Low” です。

0 : INTOUT 信号をネゲート(初期値)

1 : INTOUT 信号をアサート

WakeUp

スリープ状態解除ビット。制御マイコンにより “1” をセットすることで、スリープ処理によるスリープ状態を解除できます。

解除後、本ビットは自動的に “0” になります。

0 : 通常動作(初期値)

1 : スリープ状態解除

Rsvd2～3, 5～6 “0” 固定。

INTOUT 信号に関する説明 :

<INTOUT 信号がアサートする条件>

- ・ LSI が初期化処理を完了したときにアサートします。
 - ・ 設定された全てのチャンネルの計測処理を完了したときに毎回アサートします。
- なお、静的オフセットキャリブレーション完了時は INTOUT をアサートしません。

<INTOUT 信号がネゲートする条件>

- ・ LSI のリセットが解除されたときにネゲートします。
- ・ インターバル期間またはスリープ期間内に Control 2 Register 【Address =0x40】 の IntOut ビットに “0” を書き込むとネゲートします。
- ・ Control 1 Register 【Address=0x1C】 の WriteReq ビットと ParaCh ビットに “1” を書き込んだ後、LSI がパラメータ更新処理を完了するとネゲートします。

LC717A00 アプリケーションノート

●CDAC Offset Plus Register

Address 0x6C	Bit	7	6	5	4	3	2	1	0
	Name	CdacP7	CdacP6	CdacP5	CdacP4	CdacP3	CdacP2	CdacP1	CdacP0
	Reset	X	X	X	X	X	X	X	X
	R/W	R	R	R	R	R	R	R	R

キャリブレーションの実施により調整したオフセット値(正側)を確認するためのレジスタです。
0x00～0xFF の 8bit データです。

$$\text{オフセット容量値[fF]} (\text{Typ}) = (\text{CDAC Offset Plus Register の設定値}) \times 31.25[\text{fF}]$$

●CDAC Offset Minus Register

Address 0x6D	Bit	7	6	5	4	3	2	1	0
	Name	CdacM7	CdacM6	CdacM5	CdacM4	CdacM3	CdacM2	CdacM1	CdacM0
	Reset	X	X	X	X	X	X	X	X
	R/W	R	R	R	R	R	R	R	R

キャリブレーションの実施により調整したオフセット値(負側)を確認するためのレジスタです。
0x00～0xFF の 8bit データです。

$$\text{オフセット容量値[fF]} (\text{Typ}) = (\text{CDAC Offset Minus Register の設定値}) \times 31.25[\text{fF}]$$

■オフセット容量値の読み出し手順

本 LSI の計測は、Cin0 から Cin15 の順で行います。1つのチャネルの計測を行う際は、キャリブレーション時に決定したこのチャネル用のオフセット値をこのレジスタに取り込みます。そのため、複数のチャネルを計測する場合、各チャネルを計測するたびにオフセット値を上書きするため、本レジスタのリード値は変動してしまいます。

このため、任意のチャネルのオフセット値を確認する場合は、下記手順で読み出して下さい。

- ① Use Channel Register 【Address=0x00】 の設定により、確認したいチャネルひとつだけを有効にします。
- ② 静的オフセットキャリブレーションを実施します。
- ③ 1度計測します。
- ④ CDAC Offset Plus Register と CDAC Minus Offset Register を読み出します。
- ⑤ オフセット容量値を取得したい全てのチャネルに対して①～④を繰り返します。

●SLAVE Address Register

Address 0x7F	Bit	7	6	5	4	3	2	1	0
	Name	Rsvd7	Slave6	Slave5	Slave4	Slave3	Slave2	Slave1	SA0
	Reset	0	0	0	1	0	1	1	X
	R/W	R	R	R	R	R	R	R	R

I²C のスレーブアドレス (7bit) を確認できます。

SA0 ビットは、SA0/S0 Pin が “High” の場合 “1”、SA0/S0 Pin が Low の場合 “0” となります。

SA0 ビットの値は、SA0/S0 Pin の状態が反映されるため常に変更は可能ですが、固定して使用してください。

I²C 互換バススレーブアドレス

SA0 入力	7bit スレーブアドレス	バイナリ表記	8bit スレーブアドレス
Low	0x16	00101100b (Write)	0x2C
		00101101b (Read)	0x2D
High	0x17	00101110b (Write)	0x2E
		00101111b (Read)	0x2F

LC717A00 アプリケーションノート

1.3 推奨設定値

推奨パターンにおける推奨設定値を以下に示します。
 推奨パターンについては、デザインノート（別ドキュメント）を参照して下さい。

Reg Addr	推奨値	名称	説明
0x00	-	Use Channel Register	(※1)
0x01	0x50	Cin0 Gain Register	・ Cin0 1st Gain = 1600(最小) ・ Cin0 2nd Gain = 6 倍 (※2)
0x02	0x50	Cin1 Gain Register	・ Cin1 1st Gain = 1600(最小) ・ Cin1 2nd Gain = 6 倍 (※2)
0x03	0x50	Cin2 Gain Register	・ Cin2 1st Gain = 1600(最小) ・ Cin2 2nd Gain = 6 倍 (※2)
0x04	0x50	Cin3 Gain Register	・ Cin3 1st Gain = 1600(最小) ・ Cin3 2nd Gain = 6 倍 (※2)
0x05	0x50	Cin4 Gain Register	・ Cin4 1st Gain = 1600(最小) ・ Cin4 2nd Gain = 6 倍 (※2)
0x06	0x50	Cin5 Gain Register	・ Cin5 1st Gain = 1600(最小) ・ Cin5 2nd Gain = 6 倍 (※2)
0x07	0x50	Cin6 Gain Register	・ Cin6 1st Gain = 1600(最小) ・ Cin6 2nd Gain = 6 倍 (※2)
0x08	0x50	Cin7 Gain Register	・ Cin7 1st Gain = 1600(最小) ・ Cin7 2nd Gain = 6 倍 (※2)
0x09	0x0A	Cin0 Threshold Register	Cin0 しきい値 = 10
0x0A	0x0A	Cin1 Threshold Register	Cin1 しきい値 = 10
0x0B	0x0A	Cin2 Threshold Register	Cin2 しきい値 = 10
0x0C	0x0A	Cin3 Threshold Register	Cin3 しきい値 = 10
0x0D	0x0A	Cin4 Threshold Register	Cin4 しきい値 = 10
0x0E	0x0A	Cin5 Threshold Register	Cin5 しきい値 = 10
0x0F	0x0A	Cin6 Threshold Register	Cin6 しきい値 = 10
0x10	0x0A	Cin7 Threshold Register	Cin7 しきい値 = 10
0x1D	0x80	Average Count Register	計測データ平均回数 = 128 回
0x1E	0x0C	Filter Parameter Register	・ フィルタパラメータ 1 = 12 ・ フィルタパラメータ 2 = 0
0x1F	0x01	Debounce Count Register	デバウンスカウント = 2 回
0x20	0x05	Short Interval Time Register	ショートインターバル時間 = 5ms
0x21	0x01	Long Interval Time Register	ロングインターバル時間 = 101 (1+100) ms
0x22	0x04	Short Interval Dynamic OffCal Cycle Register	ショートインターバル時の動的オフセットキャリブレーション判定実施サイクル = 4 サイクル
0x23	0x03	Dynamic OffCal Count Plus Register	動的オフセットキャリブレーション実施カウント(正值側) = 24 回
0x24	0x03	Dynamic OffCal Count Minus Register	動的オフセットキャリブレーション実施カウント(負値側) = 3 回

次ページへ続く。

LC717A00 アプリケーションノート

前ページより続く。

Reg Addr	推奨値	名称	説明
0x25	0x32	Touch ON Cancel Count Lower Register	タッチ ON キャンセル時間カウンタ = 306 回
0x26	0x01	Touch ON Cancel Count Higher Register	
0x3D	0x40	Static OffCal CDAC Base Register	静的オフセットキャリブレーション基準容量値 = 2pF (*3)
0x3E	0x10	Measurement Mode Register	<ul style="list-style-type: none"> ・ 正値側動的オフセットキャリブレーションは全チャネルタッチ OFF 時のみ ・ タッチ OFF しきい値 = タッチ中ピーク値の 1/2 ・ ロングインターバルベース時間 = +100ms

*1 使用するチャネル数に応じて設定して下さい。

*2 デザインノート（別ドキュメント）に従って、2nd Gain を調整して下さい。

*3 デザインノート（別ドキュメント）に従って、Cin(Cref)-Cdrv 間容量に応じた設定値を調整して下さい。

2. 機能

2.1 機能説明

2.1.1 インターバルモードとスリープモード

・インターバルモード（推奨）

計測処理のあとにショートインターバル時間またはロングインターバル時間に従いインターバル処理を行うモードです。Control 1 Register 【Address=0x1C】にて選択します。

・スリープモード

計測処理のあとにスリープ処理(Deep Sleep 状態。メインクロック、ロジック回路共に停止)を行うモードです。その後、制御マイコンからの WakeUp 処理を行うことで動作が再開されます。Control 1 Register 【Address=0x1C】にて選択します。

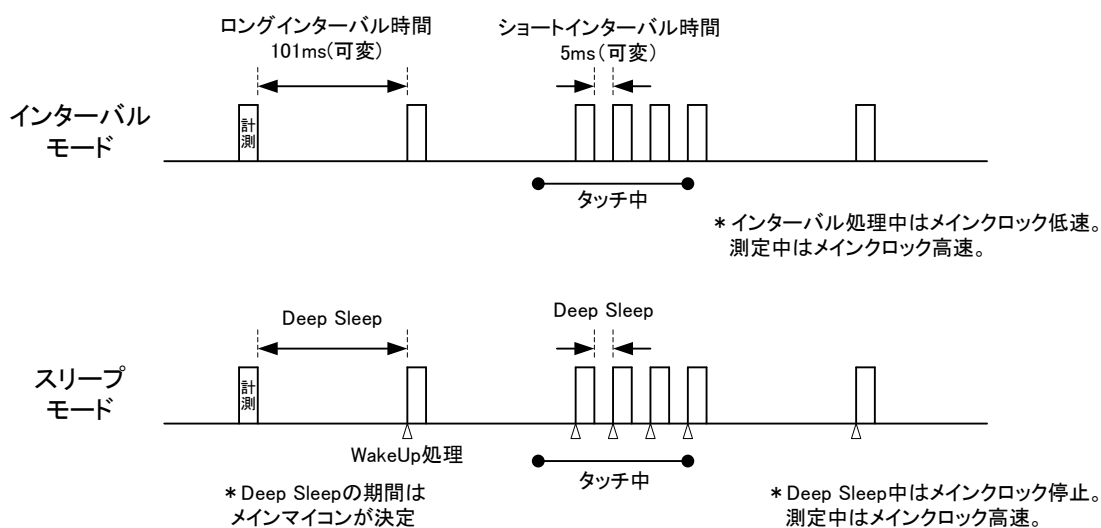


図 2-1 インターバルモードとスリープモード

2.1.2 ショートインターバルモードとロングインターバルモード

・ショートインターバルモード

インターバルモード中のインターバル処理の時間が Short Interval Time Register 【Address=0x20】により決まるモード(ショートインターバル時間)です。タッチを検出中とタッチ検出がされなくなつてから少しの間はこのモードになります。

「ショートインターバルモード」への移行条件

1つ以上のチャネルの計測データがしきい値以上だったとき、直ちに「ショートインターバルモード」に移行します。

・ロングインターバルモード

インターバルモード中のインターバル処理の時間が Long Interval Time Register 【Address=0x21】および Measurement Mode Register 【Address=0x3E】の LIVAL ビットにより決まるモード(ロングインターバル時間)です。タッチが検出されていないときにこのモードになります。

「ロングインターバルモード」への移行条件

「ショートインターバルモード」中に、計測対象チャネルの計測データの全てが Debounce Count Register 【Address=0x1F】の値の回数だけしきい値を連続して下回った後、「ロングインターバルモード」に移行します。

2.1.3 キャリブレーション

・静的オフセットキャリブレーション

静的オフセットキャリブレーションは、タッチされていない状態での計測データの値を基準値として、この基準値を計測範囲(-128~127)の中央値に合わせ込むための処理です。

静的オフセットキャリブレーションは、システム起動時または Control 1 Register 【Address=0x1C】によって設定された時に LSI 内部で自動的に実施します。

静的オフセットキャリブレーションエラー発生(基準値が中央値からの規定範囲に入らない)時には ERROR 信号がアサートし、Error Status Register 【Address=0x1A】の CALERR に“1”をセットするとともに、Error Channel Status Register 【Address=0x1B】のエラーが発生したチャンネルに対応するビットに“1”をセットします。

このとき出力 Pout0~7 では、エラーが発生したチャンネルに対応した PoutX (X=0, 1, ..., 7) に“Low”を出力します。

・動的オフセットキャリブレーション

動的オフセットキャリブレーションは、LSI 動作中の温度や湿度など外的条件による基準値のずれ(中央値である 0 からのずれ)を補正するための処理です。

動的オフセットキャリブレーションは、タッチを検出していない場合の計測時に実施し、基準値のずれが発生した場合は 0 へと収束させます。

“タッチを検出していない状態”とは、「全チャンネルの計測データがしきい値未満であるという判定が Debounce Count Register 【Address=0x1F】の値の回数以上連続している状態」です。(チャンネルの判定が OFF であっても、しきい値を超える計測データが 1 回でもあったなら、もう一度デバウンスカウンタの回数分しきい値未満の判定が続くまでは動的オフセットキャリブレーションを行いません。)

動的オフセットキャリブレーションは LSI 内部で自動的に実施します。

動的オフセットキャリブレーションエラー発生(基準値が中央値からの規定範囲に入らない)時には ERROR 信号がアサートし、Error Status Register 【Address=0x1A】の CALERR ビットに“1”をセットするとともに、Error Channel Status Register 【Address=0x1B】のエラーが発生したチャンネルに対応するビットに“1”をセットします。

このとき出力 Pout0~7 では、エラーが発生したチャンネルに対して“Low”を出力します。

エラーが発生し“Low”固定となったチャンネルが次回以降の動的オフセットキャリブレーションにて正常となった場合、そのチャンネルは通常状態に復帰します。

2.1.4 ノイズ除去

外来ノイズの耐性を上げるため、フィルタ処理によりノイズを除去しています。

この機能は LSI 内部で自動的に実行します。

2.1.5 タッチ ON 自動キャンセル機能

チャンネルのタッチ ON 判定の状態が一定回数連続して検出した場合、そのチャンネルに自動で静的オフセットキャリブレーションを実施し、タッチ ON 状態をキャンセルします。回数は Touch ON Cancel Count Lower Register 【Address=0x25】と Touch ON Cancel Count Higher Register 【Address=0x26】により変更できます。

2.2 計測機能

2.2.1 計測タイミングとロングインターバル/ショートインターバルの設定との関係

例として次の設定で動作させたときの計測タイミングの模式図を以下に示します。

- Use Channel Register 【Address=0x00】 の設定値が 0xFF (計測チャネル数 8ch)
- Average Count Register 【Address=0x1D】 の設定値が 0x40 (1 回の計測時間=13ms)
- Filter Parameter Register 【Address=0x1E】 の設定値が 0x04 (初期設定)
- Debounce Count Register 【Address=0x1F】 の設定値が 0x02 (デバウンスカウント=3 回)
- Short Interval Time Register 【Address=0x20】 の設定値が 0x07 (ショートインターバル時間=7ms)
- Long Interval Time Register 【Address=0x21】 の設定値が 0x07 (ロングインターバル時間=107ms)
- Measurement Mode Register 【Address=0x3E】 の LIVALB ビットの設定値が 0 (ロングインターバルのベース時間=100ms)

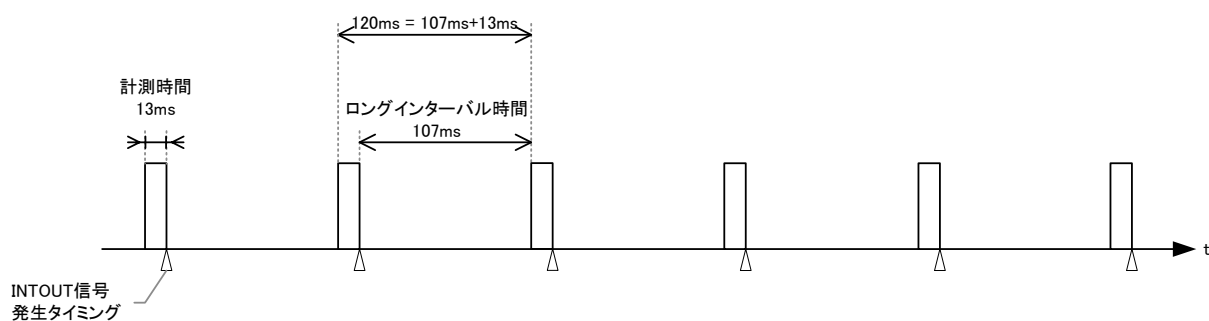
なお、以降の図においてしきい値を超えた計測点は水色の四角で示しており、しきい値を下回った計測点は白色の四角で示しています。

図内のタッチ ON/OFF ステータス CinxACT は Result Data Register 【Address=0x19】 に反映される値です。

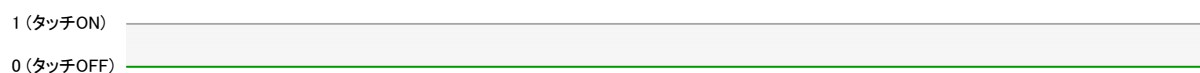
図内の全チャネルタッチなしステータス AllChNoTch は LSI が内部に保持する値です。

LC717A00 アプリケーションノート

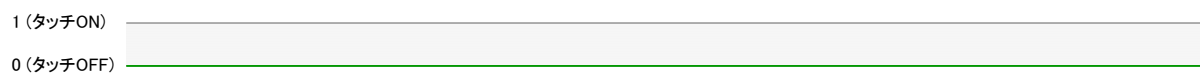
(1) 全てのチャンネルがタッチされていない状態が続いた場合の計測タイミング



Cin0 タッチON/OFFステータス Cin0ACT (LSI内部信号)

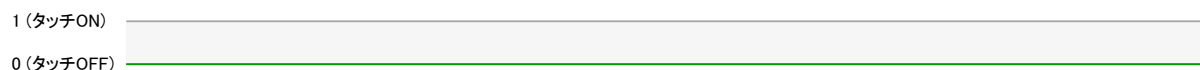


Cin1 タッチON/OFFステータス Cin1ACT (LSI内部信号)



⋮

Cin7 タッチON/OFFステータス Cin7ACT (LSI内部信号)



全チャンネルタッチなしステータス AllChNoTch (LSI内部信号)

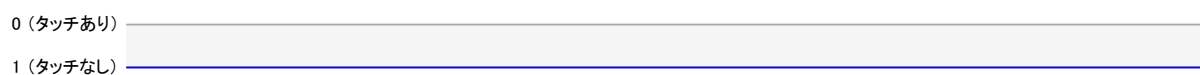


図 2-2 全てのチャンネルがタッチされていない状態が続いた場合の計測タイミング

LC717A00 アプリケーションノート

(2) 前記(1)の状態から、1つ以上のチャンネルがタッチされた時間近傍の計測タイミング

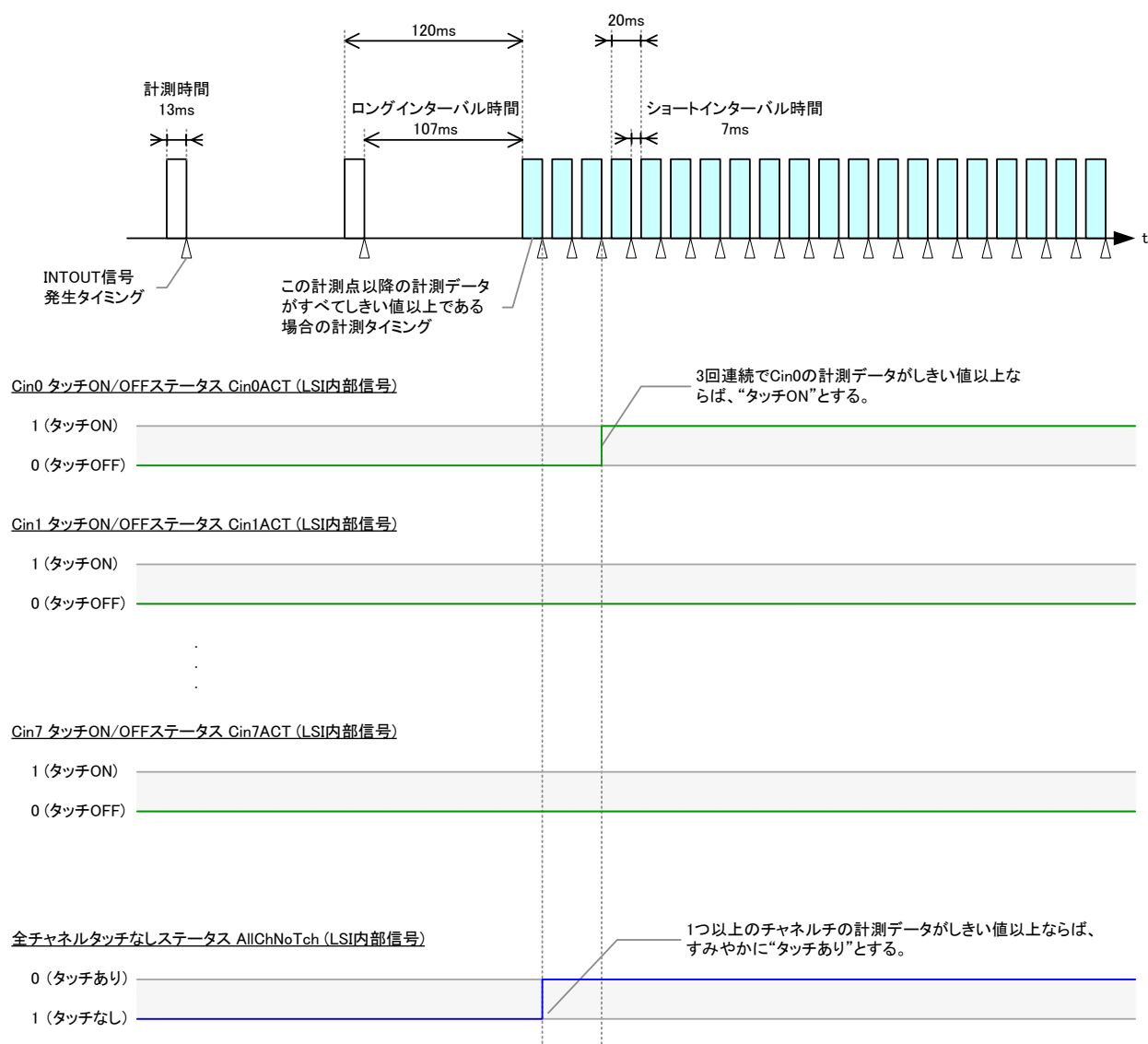
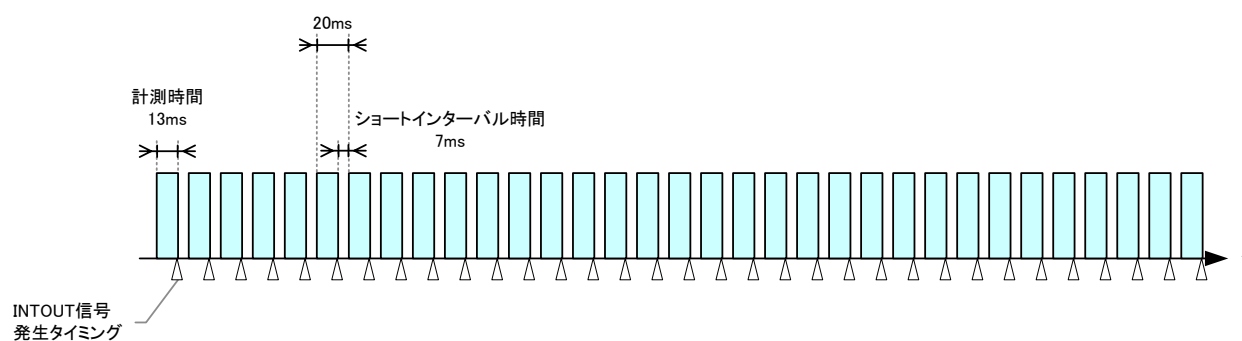


図 2-3 前記(1)の状態から、1つ以上のチャンネルがタッチされた時間近傍の計測タイミング

LC717A00 アプリケーションノート

(3) 1 つ以上のチャネルがタッチされている状態が続いた場合の計測タイミング



Cin0 タッチON/OFFステータス Cin0ACT (LSI内部信号)

1 (タッチON)

0 (タッチOFF)

Cin1 タッチON/OFFステータス Cin1ACT (LSI内部信号)

1 (タッチON)

0 (タッチOFF)

⋮

Cin7 タッチON/OFFステータス Cin7ACT (LSI内部信号)

1 (タッチON)

0 (タッチOFF)

全チャネルタッチなしステータス AllChNoTch (LSI内部信号)

0 (タッチあり)

1 (タッチなし)

図 2-4 1 つ以上のチャネルがタッチされている状態が続いた場合の計測タイミング

LC717A00 アプリケーションノート

(4) 前記(3)の状態から、全ての使用チャンネルのタッチが終了した時間近傍の計測タイミング

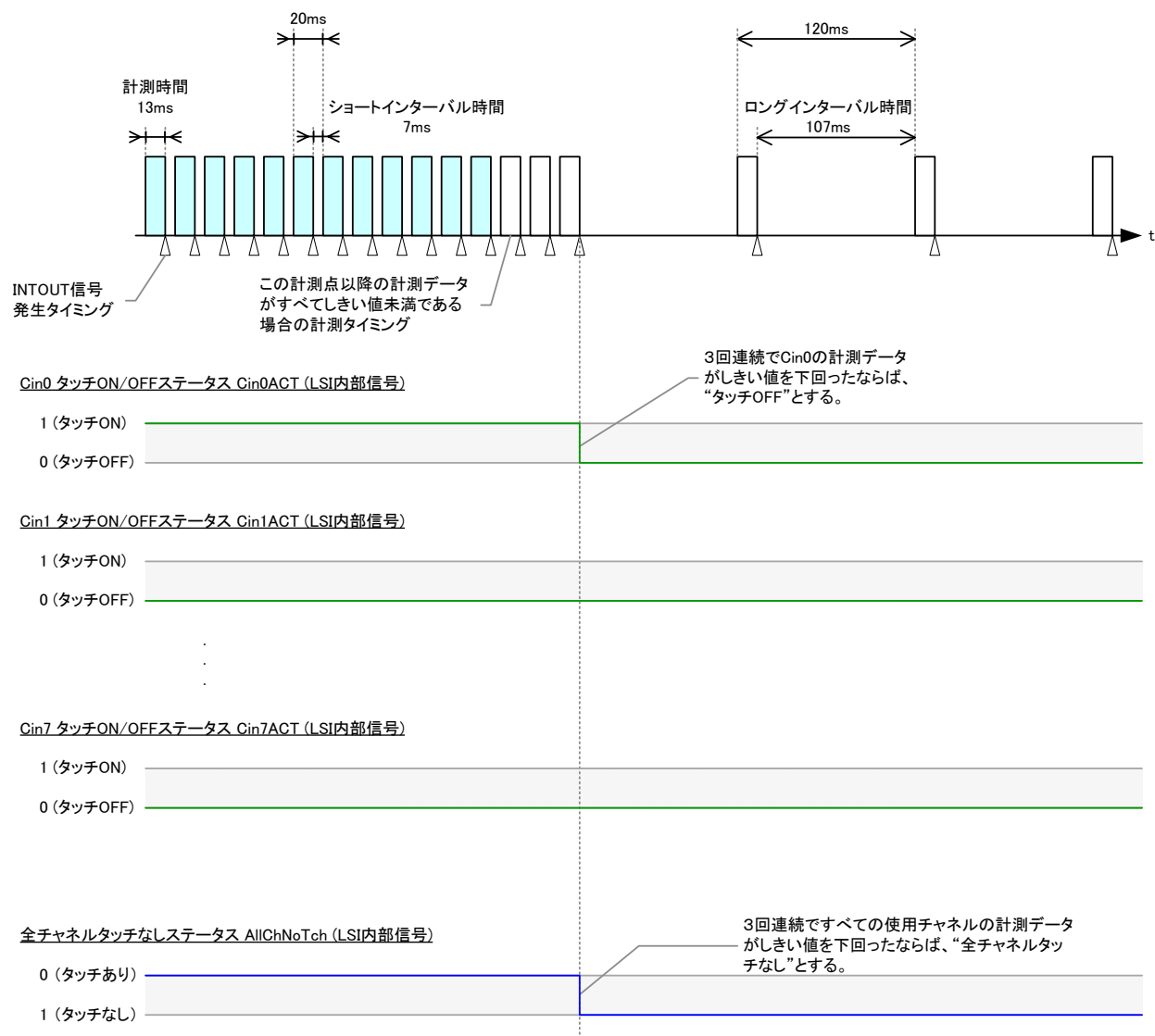


図 2-5 前記(3)の状態から、全ての使用チャンネルのタッチが終了した時間近傍の計測タイミング

LC717A00 アプリケーションノート

- (5) 前記(1)の状態から、ある計測点での計測データがしきい値を超えた場合は、最低でもデバウンスカウントで設定した回数分インターバル時間で計測を行います。そして、最初の検出以降しきい値を超えない場合は“タッチ OFF”として判定します。

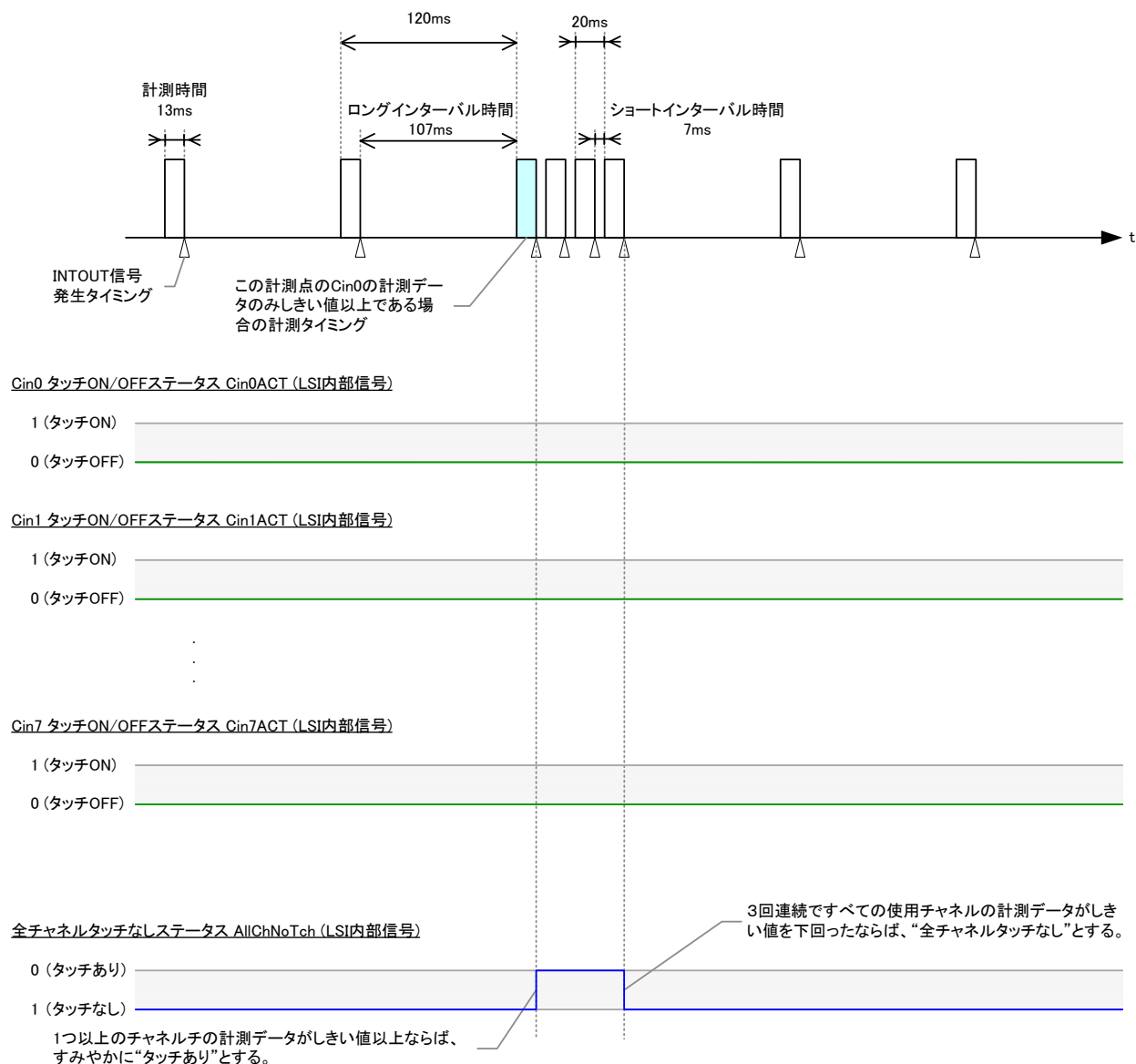


図 2-6 前記(1)の状態からある計測点での計測データのみがしきい値を超えた場合の計測タイミング

LC717A00 アプリケーションノート

(6) チャタリング等の影響を受けた場合の計測タイミング

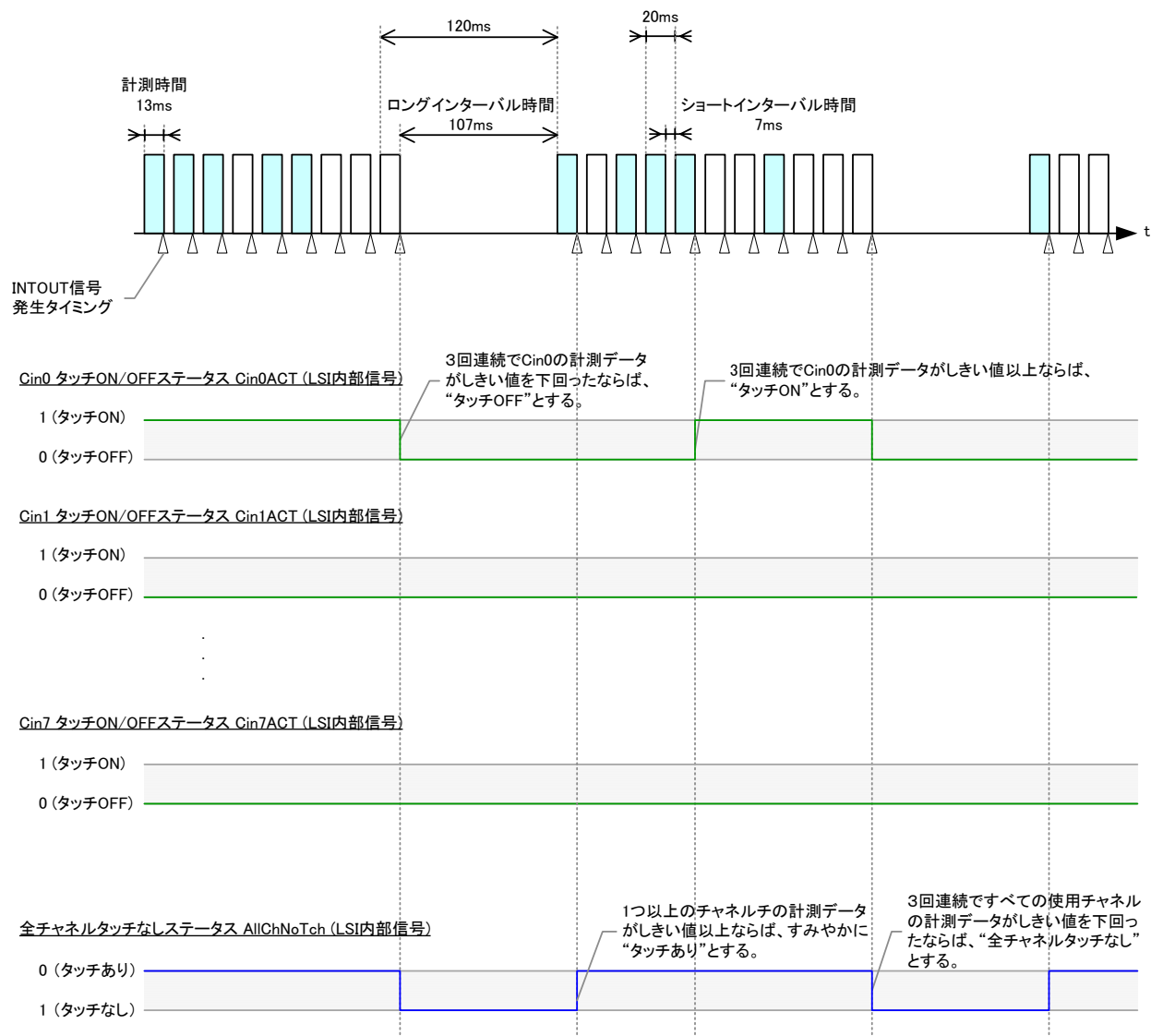


図 2-7 チャタリング等の影響を受けた場合の計測タイミング

2.2.2 動的オフセットキャリブレーションの説明

Dynamic OffCal Count Plus Register 【Address=0x23】、Dynamic OffCal Count Minus Register 【Address=0x24】で指定した回数連続して計測値が動的オフセットキャリブレーション実行範囲(4～しきい値、または、-128～-4)を取ると、動的オフセットキャリブレーションを行い計測の基準値を0に補正します。

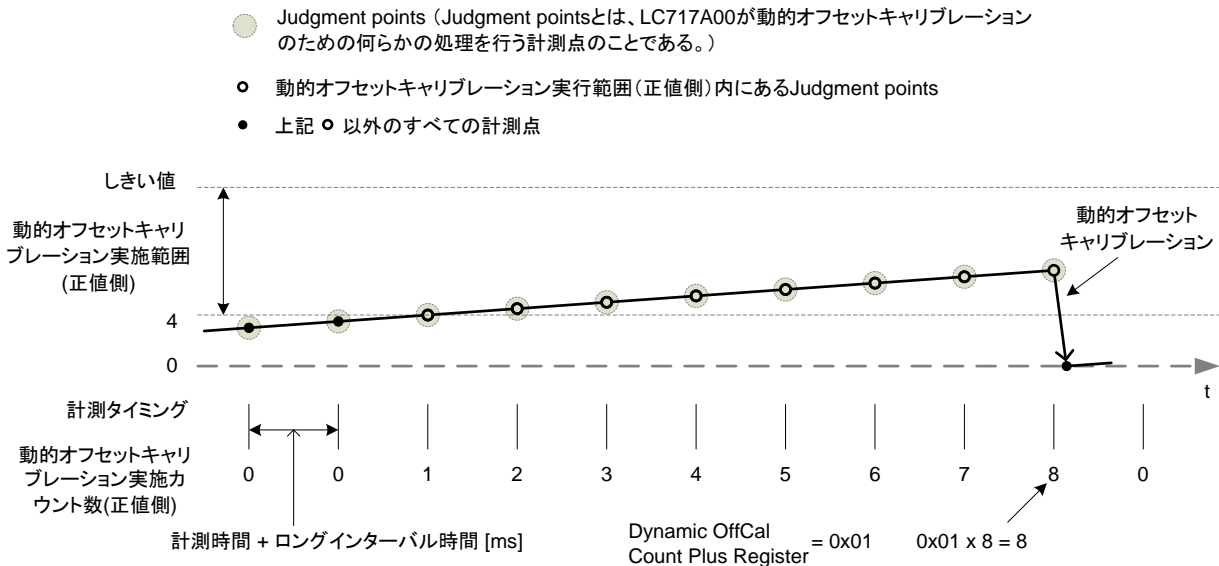
(1) ロングインターバルモードのとき

計測毎に動的オフセットキャリブレーション実施の判定を行います。

計測値が Dynamic OffCal Count Plus Register の回数連続して動的オフセットキャリブレーション実施範囲(4～しきい値)に入った場合、動的オフセットキャリブレーションを実施します。

下図は、計測値が徐々に中央値から正の方向にドリフトしていった場合の動作を示します。

Dynamic OffCal Count Plus Register=0x01(動的オフセットキャリブレーション実施カウント値=8回)のとき



※Judgment pointでの計測データが動的オフセットキャリブレーション実施範囲外になったならば、動的オフセットキャリブレーション実施カウント数(正值側)は0にリセットされる。

図 2-8 ロングインターバルモードの場合におけるあるチャンネルの動的オフセットキャリブレーション処理の一例(この場合は、すべての計測点が Judgment points となります。)

ロングインターバルモードでの動的オフセットキャリブレーションが実施されるまでの時間は以下の通りです。

(計測時間+ロングインターバル時間) × (動的オフセットキャリブレーション実施カウント値-1) [ms]

(2) ショートインターバルモードのとき

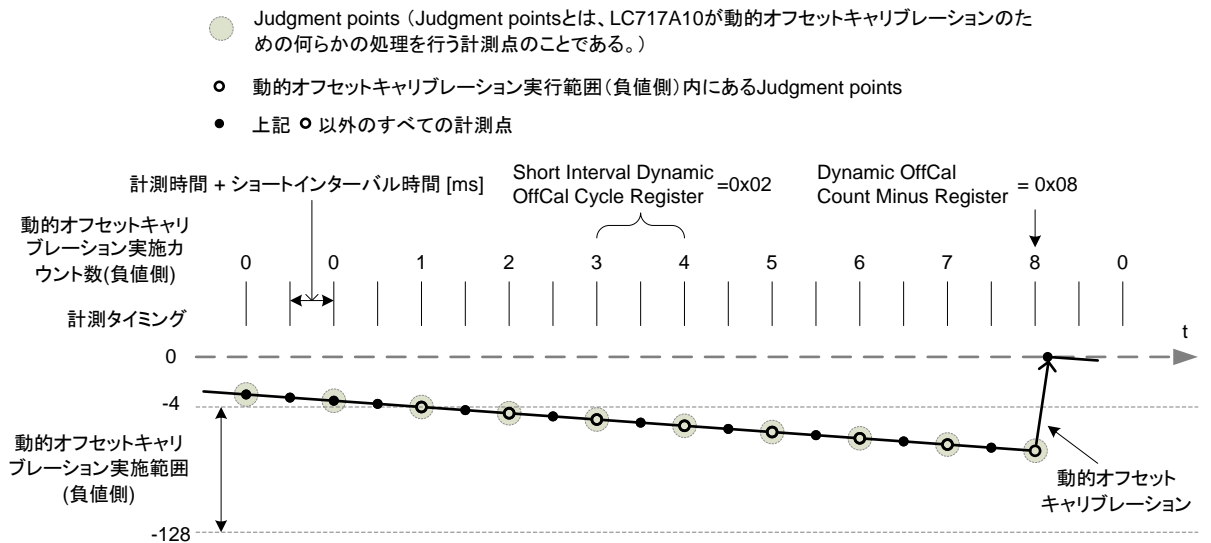
ショートインターバル時間で計測を行うため、基本的に「(1) ロングインターバルモードのとき」と比べて計測間隔が短くなるので、Short Interval Dynamic OffCal Cycle 【Address=0x22】 で指定した回数の計測毎に動的オフセットキャリブレーション判定処理を行います。

計測値が Dynamic OffCal Count Minus Register の回数連続して動的オフセットキャリブレーション実施範囲 (-128~-4) に入った場合、動的オフセットキャリブレーションを実施します。

下図は、計測値が徐々に中央値から負の方向にドリフトしていった場合の動作を示します。

Dynamic OffCal Count Minus Register=(動的オフセットキャリブレーション実施カウント値=8 回)のとき

Short Interval Dynamic OffCal Cycle Register=0x02(動的オフセットキャリブレーション判定実施サイクル=2 回)のとき



※Judgment pointでの計測データが動的オフセットキャリブレーション実施範囲外になったならば、動的オフセットキャリブレーション実施カウント数(負値側)は0にリセットされる。

図 2-9 ショートインターバルモードの場合における
あるチャンネルの動的オフセットキャリブレーション処理の一例
(この場合は、Short Interval Dynamic OffCal Cycle Register の
設定値に従って、一部の計測点のみが Judgment points となります。)

ショートインターバルモードでの動的オフセットキャリブレーションが実施されるまでの時間は以下の通りです。

(計測時間+インターバル時間) × (動的オフセットキャリブレーション実施カウント値-1)
× 動的オフセットキャリブレーション判定実施サイクル数 [ms]

2.2.3 計測間隔の概算式

使用チャネル数が 8ch のときの計測間隔(計測開始から次の計測開始までの間隔)の概算式

$$\begin{aligned} & \text{ショートインターバルモードの計測間隔[ms] (Typ)} \\ &= \frac{\{\text{FP1k} \times (\text{Average Count Register の設定値} + 4)\}}{\text{1ch の計測処理に掛かる時間}} \times \underbrace{8}_{\text{計測する ch 数}} + \underbrace{0.80}_{\text{LSI 内部処理時間}} + (\text{ショートインターバル時間}) \end{aligned}$$

$$\begin{aligned} & \text{ロングインターバルモードの計測間隔[ms] (Typ)} \\ &= \{\text{FP1k} \times (\text{Average Count Register の設定値} + 4)\} \times 8 + 0.80 + (\text{ロングインターバル時間}) \end{aligned}$$

FP2[3:0]=0の時のFP1kはFilter Parameter Register (FP1[3:0])の値により以下の式で求められます。

$$\text{FP1k} = 0.021 + (\text{FP1}[3:0] \times 0.000375)$$

使用チャネル数が 8ch 且つ初期設定(計測データ平均回数 64 回、FP1[3:0]=4)、ショートインターバル時間=5[ms]、ロングインターバル時間=101[ms])での計測間隔は、次のようになります。

ショートインターバルモードの計測間隔 = 18.0[ms] (Typ)

ロングインターバルモードの計測間隔 = 114.0[ms] (Typ)

2.2.4 タッチ ON 自動キャンセル機能の説明

タッチ ON/OFF ステータス CinxACT ビットの“タッチ ON”状態が一定期間続くと、そのチャネルに対して静的オフセットキャリブレーションを実施し、“タッチ ON”をキャンセルします。

但し、実際にタッチが OFF になるのは、デバウンスカウント分経過してからになります。

下図は、Debounce Count Register 【Address=0x1F】が 0x02(3 回)、Touch ON Cancel Count Register 【Address=0x25/0x26】が 0x0005(タッチ ON 自動キャンセルカウント数=5 回)の場合を示します。

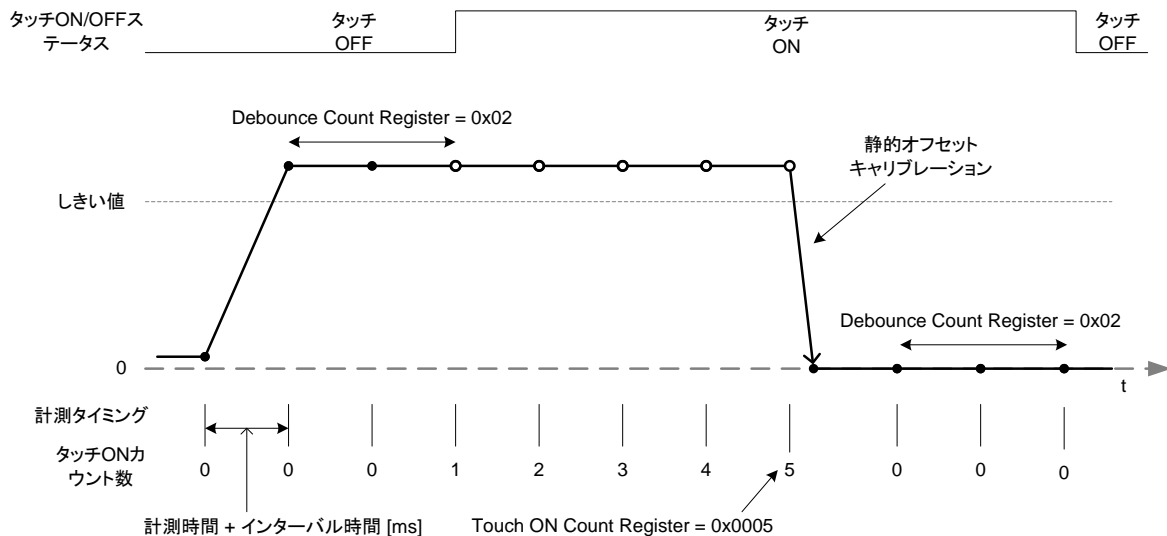


図 2-10 タッチ ON 自動キャンセル機能

タッチ ON してから静的オフセットキャリブレーションを実施するまでの時間は、以下の通りです。

$$(\text{計測時間} + \text{インターバル時間}) \times (\text{タッチ ON 自動キャンセルカウント数} - 1) [\text{ms}]$$

2.2.5 タッチ OFF 判定時の動的しきい値制御の説明

LC717A00 は、タッチ ON 状態からタッチ OFF 状態に遷移する際に適用するしきい値として、動的しきい値を使用します。

Measurement Mode Register 【Address=0x3E】の TOFFTH ビットを 0 に設定したときの動的しきい値は、DYNTHR1 の値と設定しきい値 CinX Threshold Register 【Address=0x09-0x10】のうちの大きい方を適用します。ここで、DYNTHR1 は

$$\text{DYNTHR1} = \text{MAXDATA} - (\text{MAXDATA}/4) = (3/4)\text{MAXDATA}$$

であり、MAXDATA は、タッチ ON 判定になった計測点から前回の計測点までの範囲における計測データの最大値を表します。

一方、Measurement Mode Register 【Address=0x3E】の TOFFTH ビットを 1 に設定したときの動的しきい値は、DYNTHR2 の値と設定しきい値 CinX Threshold Register 【Address=0x09-0x10】のうちの大きい方を適用します。ここで、DYNTHR2 は

$$\text{DYNTHR2} = \text{MAXDATA} - (\text{MAXDATA}/2) = (1/2)\text{MAXDATA}$$

であり、MAXDATA はタッチ ON 判定になった計測点から前回の計測点までの範囲における計測データの最大値を表します。

※通常は、TOFFTH ビットを 1 に設定して使用してください。

(1) TOFFTH ビットが 1 の場合

例として、

- ・計測時間+ショートインターバル時間の値が 20msec、
- ・計測時間+ロングインターバル時間の値が 120msec、
- ・デバウンスカウントが 3、
- ・設定しきい値が 15、
- ・Measurement Mode Register 【Address=0x3E】の TOFFTH ビットが 1

の設定で LC717A00 をインターバルモードで動作させた場合におけるタッチ OFF 判定動的しきい値の動作を説明するための模式図を図 2-11 に示します。ただし、最初は、LC717A00 のすべてのチャンネルにおいてしばらくの間タッチが検出されていなかったものとします。

図 2-11 において、タッチ ON 判定用しきい値は、「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」です。一方、タッチ OFF 判定用しきい値は、「タッチ ON 判定となった計測点以降から前回の計測点までの計測データの最大値の 1/2」と「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」のうちの大きい方の値です。

Measurement Mode Register 【Address=0x3E】の TOFFTH ビットが 1 の場合において、あるチャンネルのタッチ ON/OFF 状態がタッチ ON 状態からタッチ OFF 状態に遷移するためには、「タッチ ON 判定となった計測点以降から前回の計測点までの計測データの最大値の 1/2」と「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」のうちの大きい方の値未満の計測データが指定したデバウンスカウント回連続して続いていることが必要です。

(2) TOFFTH ビットが 0 の場合

- ・計測時間 + ショートインターバル時間の値が 20msec、
- ・計測時間 + ロングインターバル時間の値が 120msec、
- ・デバウンスカウントが 3、
- ・設定しきい値が 15、
- ・Measurement Mode Register 【Address=0x3E】 の TOFFTH ビットが 0

の設定で LC717A00 をインターバルモードで動作させた場合におけるタッチ OFF 判定動的しきい値の動作を説明するための模式図を図 2-12 に示します。ただし、最初は、LC717A00 のすべてのチャンネルにおいてしばらくの間タッチが検出されていなかったものとします。

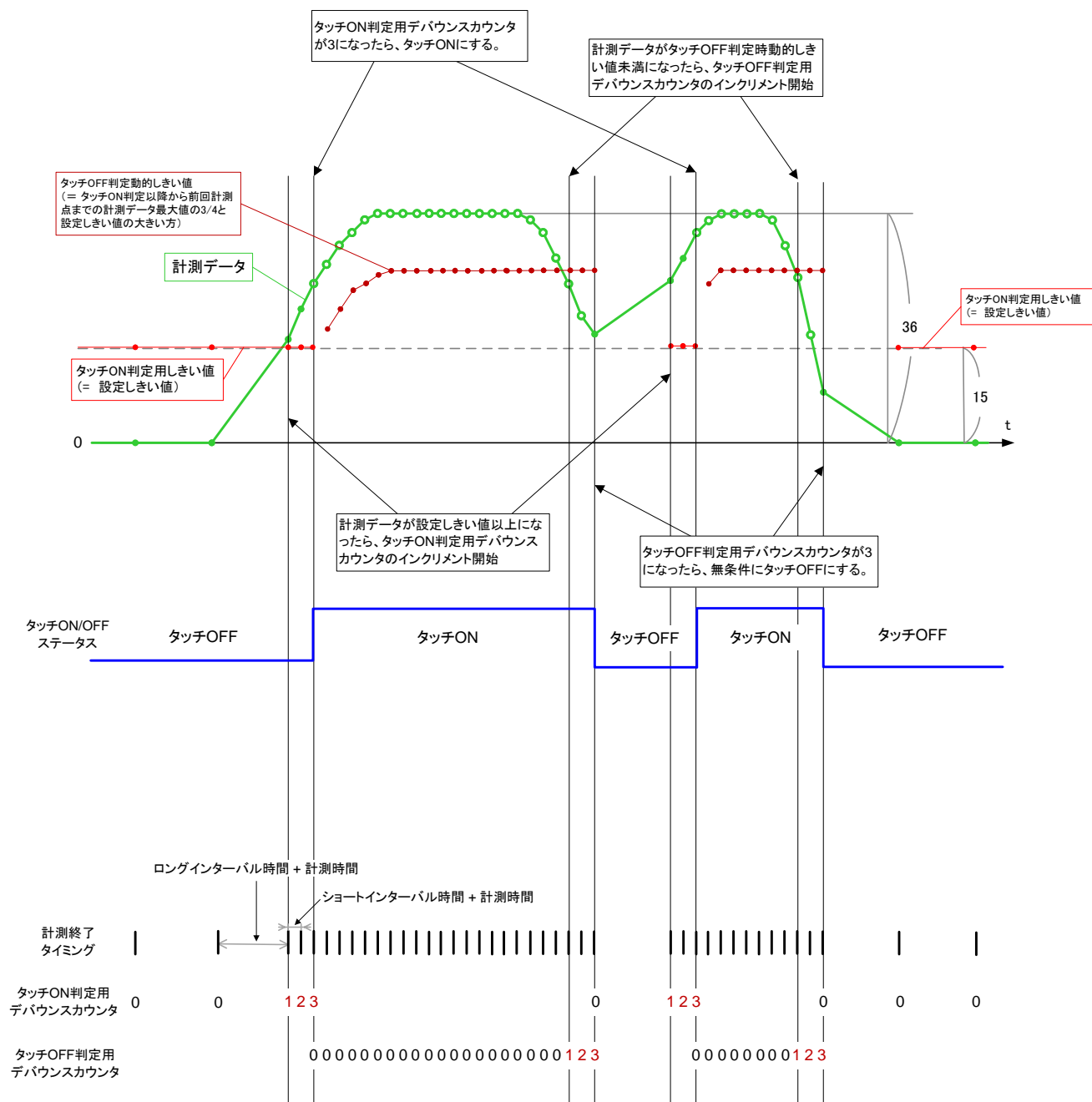


図 2-12 TOFFTH ビットが 0 の場合の例

図 2-12 において、タッチ ON 判定用しきい値は、「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」です。一方、タッチ OFF 判定用しきい値は、「タッチ ON 判定となった計測点以降から前回の計測点までの計測データの最大値の 3/4」と「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」のうちの大きい方の値です。

Measurement Mode Register 【Address=0x3E】の TOFFTH ビットが 0 の場合において、あるチャンネルのタッチ ON/OFF 状態がタッチ ON 状態からタッチ OFF 状態に遷移するためには、「タッチ ON 判定となった計測点以降から前回の計測点までの計測データの最大値の 3/4」と「CinX Threshold Register 【Address=0x09-0x10】で設定した設定しきい値」のうちの大きい方の値未満の計測データが Debounce Count Register 【Address=0x1F】で指定したデバウンスカウント回連続して続くことが必要です。

【注意】 LC717A00 はタッチ OFF 判定時に動的しきい値制御を行っています。このため、外来ノイズ等のために計測データに比較的大きなノイズを含む場合、図 2-12 に示すように、LC717A00 が出力するタッチ ON/OFF ステータスにおいて瞬間的に不要なタッチ OFF ステータスが出力される現象が発生する場合があります。TOFFTH ビットが 0 の場合は、TOFFTH ビットを 1 に設定するとこの現象が発生しなくなる可能性があります。

本 LSI は、スイッチをしっかりとタッチして ON 判定になった後の ON 判定中期間における最大計測値の 1/2 以下でかつしきい値以上の計測値が得られるタッチ状態が続くような状況を想定していません。このような状況が想定される場合には、制御マイコンを使って LC717A00 の計測データ値をリードし、制御マイコン側で LC717A00 からの計測データ値を使って ON/OFF 判定することで解決することができます。

LC717A00 アプリケーションノート

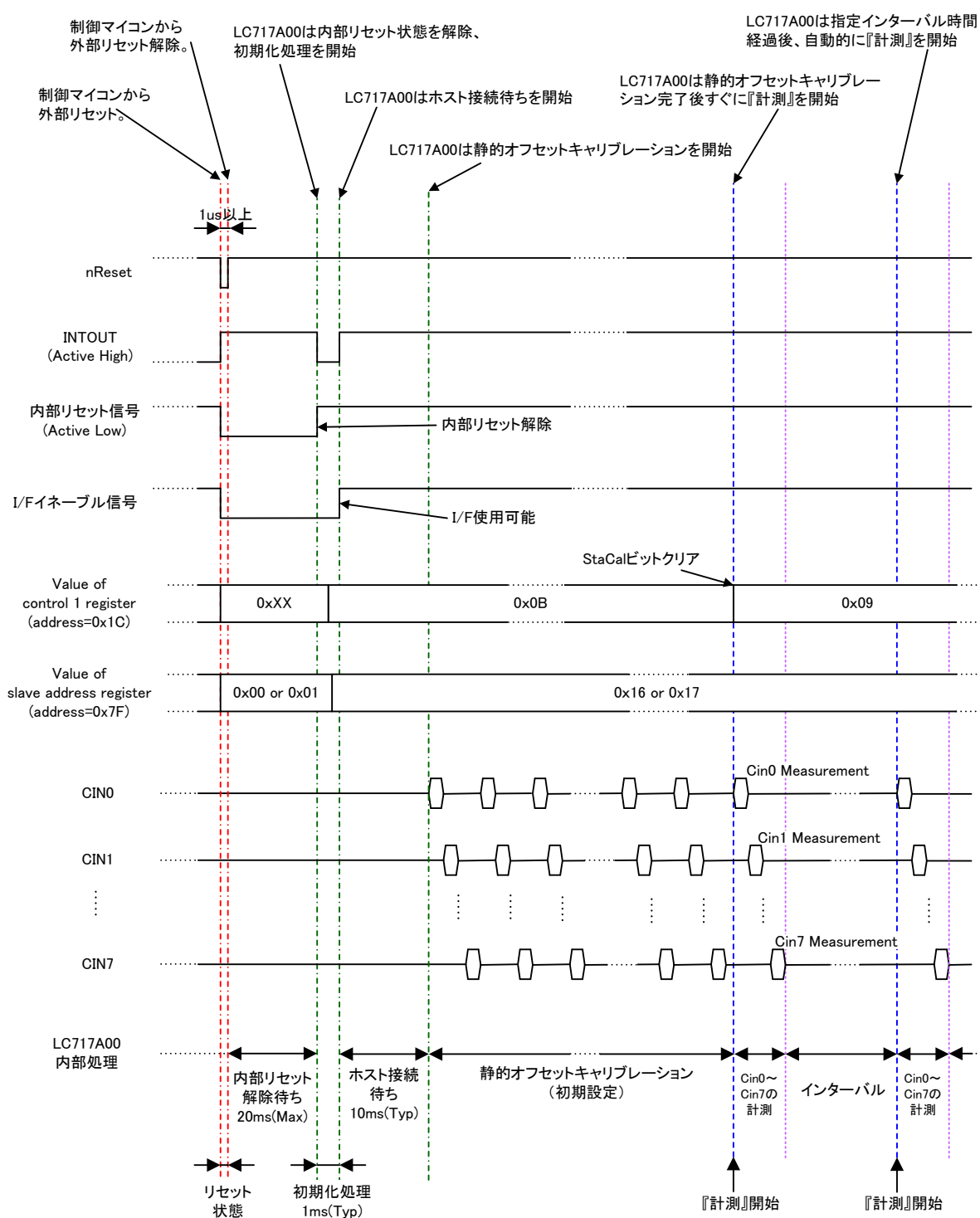


図3-1 外部リセット時の動作シーケンス例

3.2 インターバルモード時の動作シーケンス例

LC717A00を常にインターバルモードに設定した場合の動作シーケンス例を図3-2と図3-3に示します。（この動作シーケンス例は、8チャンネルのすべてを使用すると設定した場合の例です。）

(通常、LC717A00はスリープモードではなくインターバルモードで動作させます。)

図3-2と図3-3の動作シーケンスの要約は以下の通りです。

- ・ LC717A00を既にインターバルモードで動作させていたとします。
具体的には、最初、LC717A00はインターバルモードで繰り返し『計測』している状態であり、制御マイコンはLC717A00から計測結果を読み出している状態であるとしてします。
- ・ 制御マイコンは、LC717A00に新しい設定をライトした後、パラメータ変更要求および静的オフセットキャリブレーション要求を一回発行します。
- ・ LC717A00でのパラメータ変更および静的オフセットキャリブレーションが完了した後、制御マイコンはLC717A00から計測結果を読み出します。

図3-2と図3-3の動作シーケンスの詳細を以下に説明します。

- (1) LC717A00を既にインターバルモードで動作させていたとします。
具体的には、最初、LC717A00は繰り返し『計測』している状態であり、制御マイコンはINTOUTがアサートしたこと（INTOUTが”Low”から”High”に変化したこと）を検出する度にINTOUTをネゲートする（INTOUTを”Low”にする）とともにLC717A00から計測結果を読み出している状態であるとしてします。
- (2) 制御マイコンは、LC717A00内のレジスタ【Address=0x00～0x10, 0x1D～0x26, 0x3D～0x3E】のすべてまたは一部に新しい設定パラメータを書き込みます。
- (3) 制御マイコンは、Control 1 Register【Address=0x1C】に0x8Fをライトします。これにより、制御マイコンは、LC717A00に対して新しい設定をその動作に反映するとともに新しい設定に基づいて静的オフセットキャリブレーションを行うことを要求します。
- (4) LC717A00は、'Interval'の期間が終わるまでこの要求をペンディングします。
- (5) 'Interval'の期間が終了するとすぐに、LC717A00は上記(3)の要求を受け付けます。（LC717A00のControl 1 Register【Address=0x1C】のWriteReqビットが0になります。）続いて、LC717A00は、新しい設定パラメータをLC717A00内部に反映します。以降、LC717A00はこの新しい設定に基づいた動作を行います。
- (6) LC717A00は、新しい設定に基づいて静的オフセットキャリブレーションを行います。
- (7) LC717A00は、『計測』を開始します。指定したすべてのCinの『計測』が終わると、LC717A00はINTOUTをアサートします。
- (8) LC717A00からのINTOUTがアサートする毎に、制御マイコンはINTOUTをネゲートしLC717A00から計測結果を読み出します。



LC717A00 アプリケーションノート

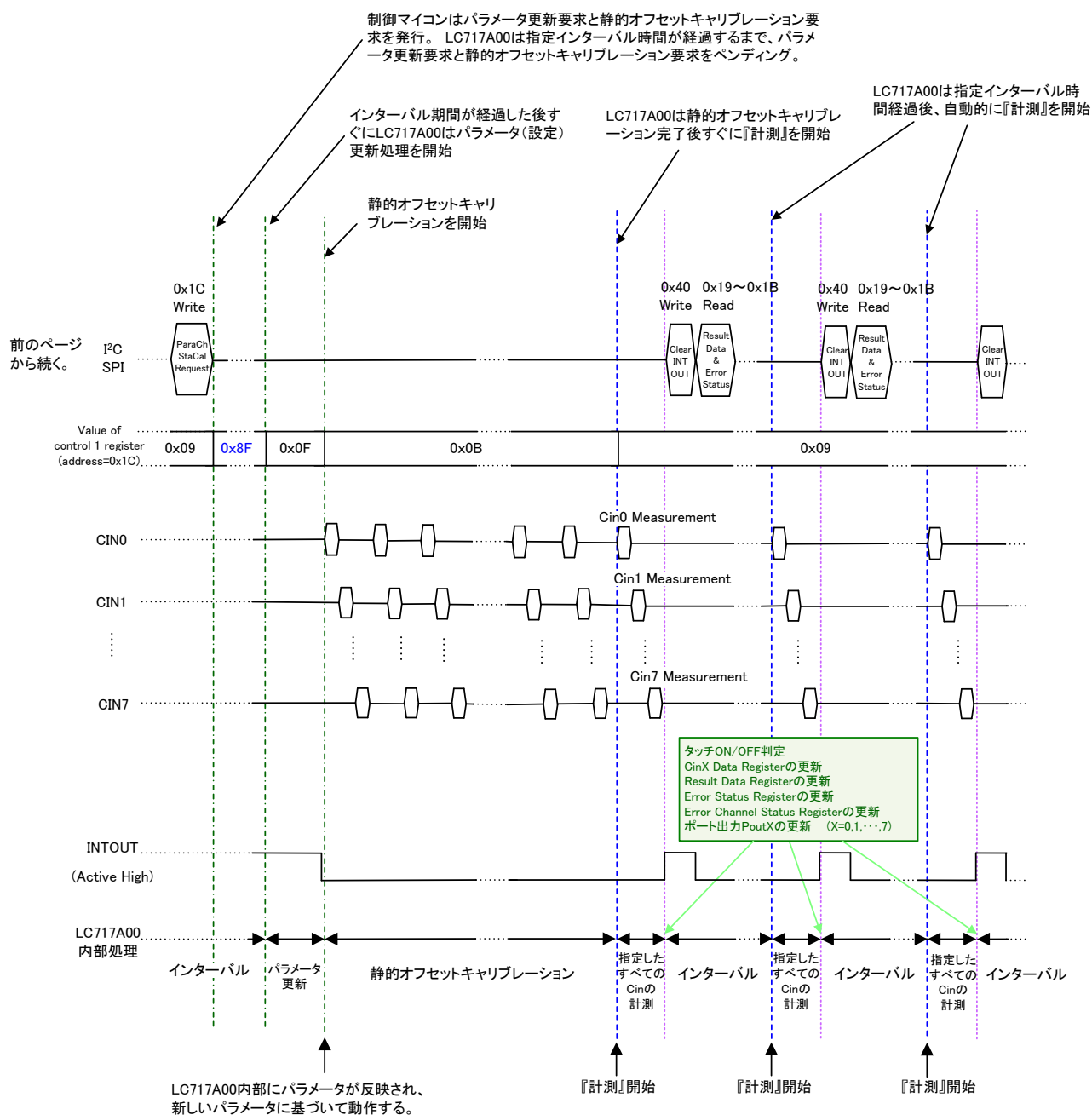


図3-3 インターバルモード動作時の動作シーケンス例 (2/2)

3.3 スリープモード時の動作シーケンス例

LC717A00を常にスリープモードに設定した場合の動作シーケンス例を図3-4と図3-5に示します。
(この動作シーケンス例は、8チャンネルのすべてを使用すると設定した場合の例です。)

スリープモードの場合、制御マイコンがスリープ状態にあるLC717A00をウェークアップすると、LC717A00は指定したすべてのCinの『計測』を1度行った後にスリープします。なお、LC717A00は、スリープする際にINTOUTをアサートします。

図3-4と図3-5の動作シーケンスの要約は以下の通りです。

- ・ LC717A00を既にスリープモードで動作させていたとします。
具体的には、最初、制御マイコンは、INTOUTがアサートしたことを検出するとLC717A00の計測結果を読み出し、所定の時間になったらLC717A00をウェークアップすることを繰り返している状態にあるとします。
- ・ 制御マイコンは、LC717A00に新しい設定をライトした後、パラメータ変更要求および静的オフセットキャリブレーション要求を一回発行します。そして、LC717A00をウェークアップします
- ・ LC717A00は、パラメータ変更と静的オフセットキャリブレーションが完了した後『計測』を行います。『計測』が終わった後、LC717A00はスリープします。
- ・ 以降、制御マイコンはLC717A00がスリープしたことを検出する度に、LC717A00から計測結果を読み出し、所定の時間になったらLC717A00をウェークアップします。

図3-4と図3-5の動作シーケンスの詳細を以下に説明します。

- (1) LC717A00を既にスリープモードで動作させていたとします。
具体的には、最初、制御マイコンはINTOUT信号を使ってLC717A00がスリープしたことを検出してLC717A00から計測結果を読み出し、所定の時間になったらLC717A00をウェークアップすることを繰り返している状態にあるとします。
- (2) 制御マイコンは、LC717A00のレジスタ【Address=0x00～0x10, 0x1D～0x26, 0x3D～0x3E】のすべてまたは一部に新しい設定パラメータを書き込みます。
- (3) 制御マイコンは、Control 1 Register【Address=0x1C】に0x87をライトします。これにより、制御マイコンは、LC717A00に対して新しい設定をその動作に反映するとともに新しい設定に基づいて静的オフセットキャリブレーションを行うことを要求します。
- (4) 制御マイコンは、LC717A00をウェークアップします。
- (5) ウェークアップ後すぐに、LC717A00は上記(3)の要求を受け付けます。(LC717A00のControl 1 Register【Address=0x1C】のWriteReqビットが0になります。) 続いて、LC717A00は新しい設定パラメータをLC717A00内部に反映します。以降、LC717A00はこの新しい設定に基づいた動作を行います。
- (6) LC717A00は、新しい設定に基づいて静的オフセットキャリブレーションを行います。

LC717A00 アプリケーションノート

- (7) LC717A00は、『計測』を開始します。指定したすべてのCinの『計測』が終わると、LC717A00はINTOUTをアサートしてスリープします。
- (8) 制御マイコンはINTOUTがアサートしたことを検出した後、LC717A00から計測結果を読み出します。そして所定の時刻になったら、制御マイコンはLC717A00をウェークアップします。
- (9) 以降、上の(7)と(8)を繰り返します。

LC717A00 アプリケーションノート

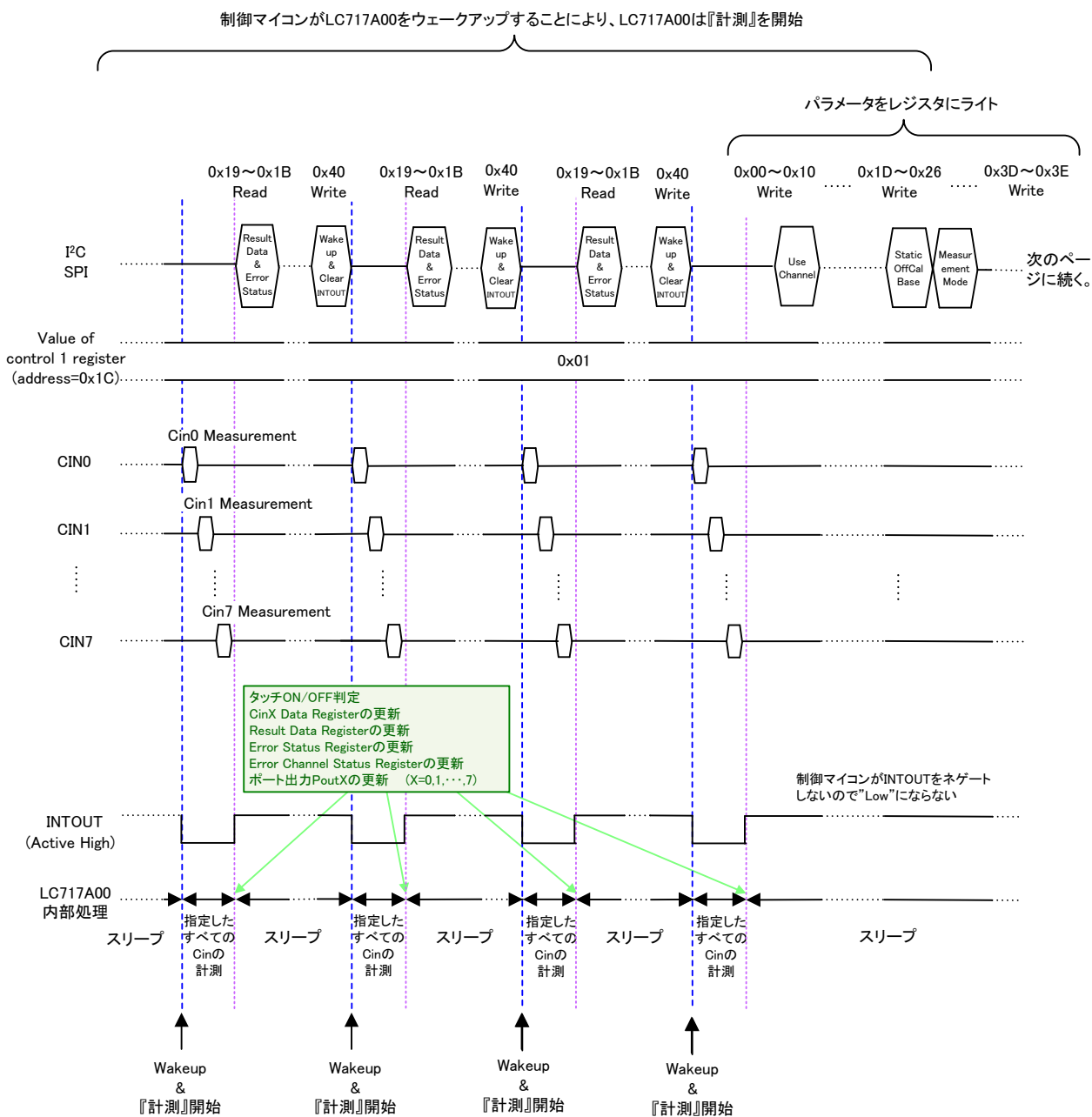


図3-4 スリープモード動作時の動作シーケンス例 (1 / 2)

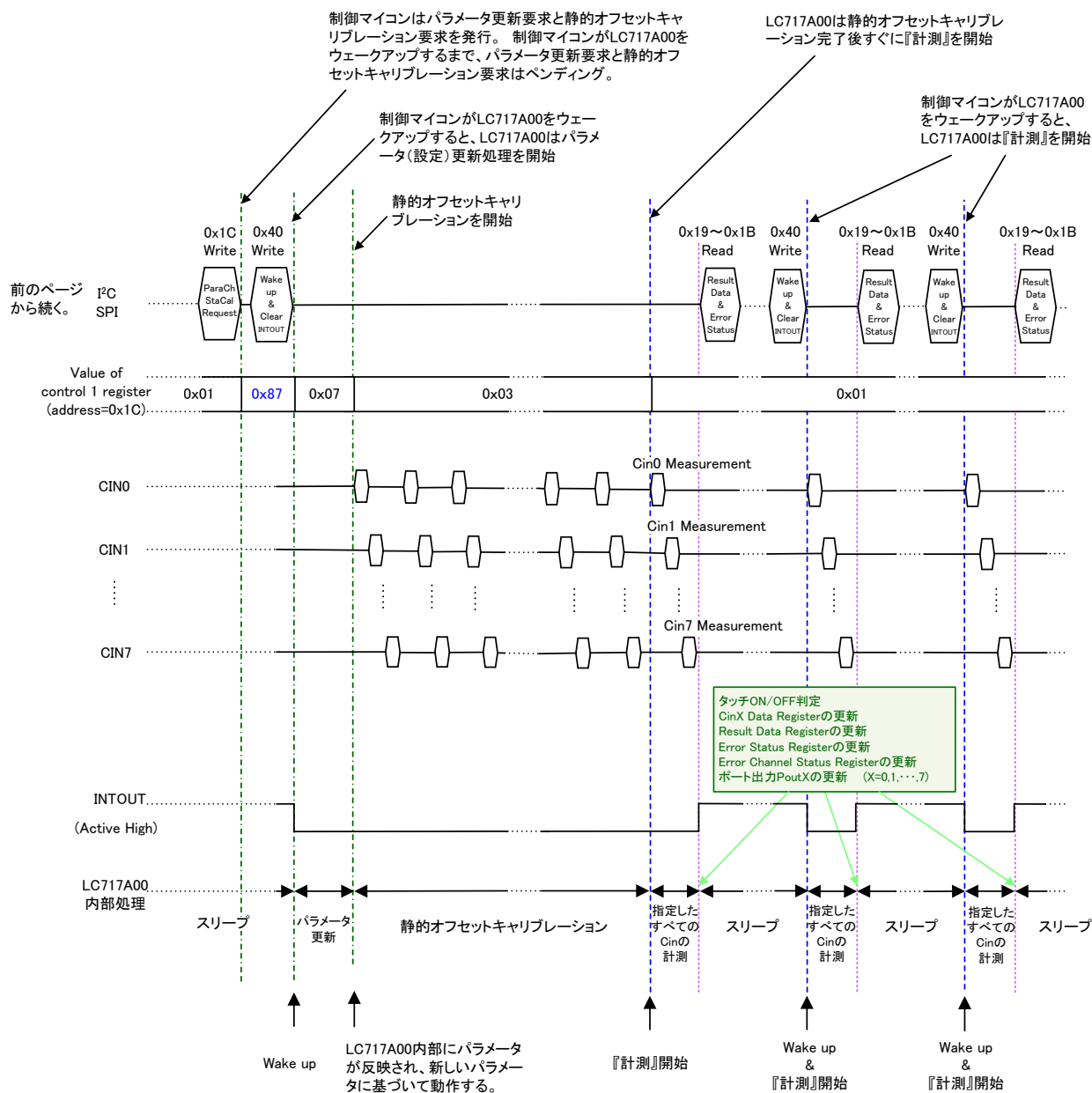


図3-5 スリープモード動作時の動作シーケンス例 (2 / 2)

4. 制御マイコンによる制御例

※本章において、『計測』処理とは、「Use Channel Register 【Address=0x00】で計測すると指定したすべてのチャンネルの計測」だけでなく「各チャンネルのタッチ ON/OFF 判定」を含んだ処理と定義します。

4.1 制御フロー例

4.1 では、制御マイコンによる制御例を以下の 4 つの場合について説明します。

- ・インターバルモードに設定し、INTOUT を使って計測結果をリードする場合
- ・インターバルモードに設定し、INTOUT を使わずに計測結果をリードする場合
- ・スリープモードに設定し、INTOUT を使って計測結果をリードする場合
- ・インターバルモードに設定し、INTOUT を使わずに PoutX (X=0, 1, ..., 7) をリードする場合

4.1.1 インターバルモードに設定し、INTOUT を使って計測結果をリードする場合

制御マイコンと LC717A00 を I²C 互換バス（または SPI バス）で接続し、制御マイコンの GPIO ポートと LC717A00 の INTOUT ピンを接続し、制御マイコンの別の GPIO ポートと LC717A00 の nRST ピンを接続した場合を考えます。このような場合において、制御マイコンが INTOUT 信号の状態を検出して『計測』完了タイミング毎にレジスタから計測結果を読み出す場合の制御例を説明します。

なお、本制御例では、制御マイコンは LC717A00 を常にインターバルモードに設定します。

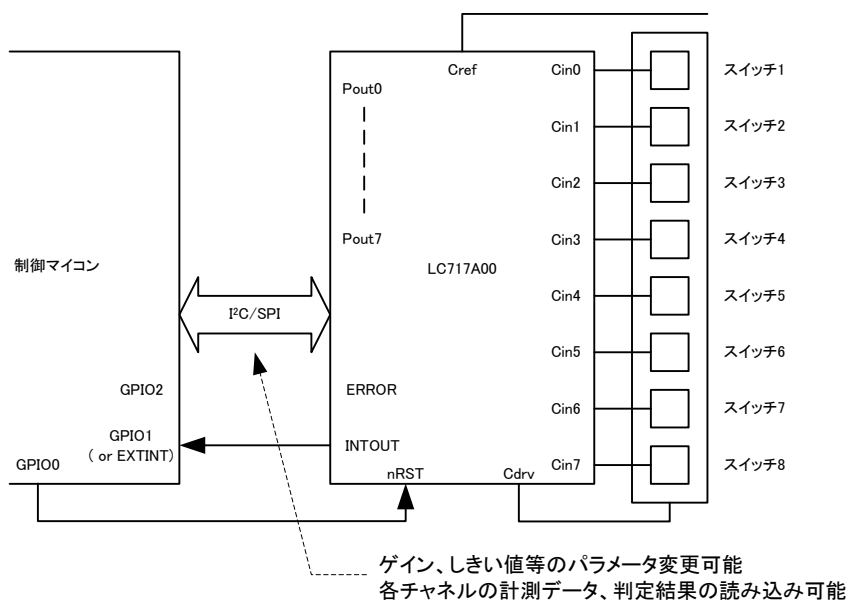


図 4-1 制御マイコンと LC717A00 との接続

- (1) 制御マイコンは、汎用出力ポートを使って nRST 信号を制御することにより LC717A00 をリセットします。

※具体的には、制御マイコンは、LC717A00 の nRST に接続した汎用出力ポートを使って nRST を”Low”にした後再び”High”にします。なお、最低 1 usec は”Low”にします。

- (2) 制御マイコンは、nRST を使った LC717A00 のリセット制御終了直後から汎用入力ポートを使って INTOUT 信号の状態を検出します。制御マイコンは、LC717A00 リセット解除後に発生する INTOUT 信号の立ち上がりエッジを検出したら、すみやかに Control 1 Register 【Address=0x1C】に 0x88 をライトします。(具体的には、LC717A00 リセット解除後に発生する INTOUT 信号の立ち上がりエッジから **5msec** 以内に 0x88 をライトします。) これを行うのは、リセット後に LC717A00 が自動実行する「静的オフセットキャリブレーションの実行と『計測』の実行」を停止するためです。
- (3) 制御マイコンは、新しい設定パラメータをレジスタ群にライトします。(具体的には、制御マイコンは、図 4-2 の青の点線で囲まれたボックス(A)内のすべてのレジスタまたは一部のレジスタに新しいパラメータをライトします。)
※なお、この時点では、LC717A00 の内部にはまだ新しい設定パラメータの変更が反映されないことに注意してください。
- (4) 新しい設定を LC717A00 内部に反映して静的オフセットキャリブレーションと『計測』をスタートするために、制御マイコンは Control 1 Register 【Address=0x1C】に 0x8F をライトします。
- (5) すると、LC717A00 は新しい設定を LC717A00 内部に反映します。
※なお、新しい設定のLC717A00内部への反映完了時に、LC717A00のINTOUT信号は自動的にネゲートします(“Low” になります)。くわえて、LC717A00は、ERRORピンへの出力を“Low” に初期化します。また、出力ポートPout0～Pout7の出力信号をすべて“Low” に初期化します。
また、LC717A00は、CinX Data Register 【Address=0x11～0x18】、Error Status Register 【Address=0x1A】、Error Channel Status Register 【Address=0x1B】のすべてのレジスタを 0x00に初期化するとともに、すべてのチャネルのタッチON/OFF判定結果をOFFに初期化することに注意してください。くわえて、動的オフセットキャリブレーション処理用内部カウンタ、デバウンス処理用内部カウンタ、タッチON自動キャンセル処理用内部カウンタの値を0クリアすることに注意してください。
- (6) 次に、LC717A00 は新しい設定で静的オフセットキャリブレーションを実行します。そして、LC717A00 はこの静的オフセットキャリブレーション完了後に『計測』を開始します。
- (7) LC717A00 の『計測』が完了したことを確認するために、制御マイコンは INTOUT 信号のアサート(INTOUT 信号が“High” になったこと)を確認します。この後、制御マイコンは、INTOUT 信号がアサートしたことを確認後、Control 2 Register 【Address=0x40】へ 0x00 をライトして INTOUT 信号をネゲートし、Result Data Register 【Address=0x19】等のレジスタから計測結果を読み出します。
- (8) LC717A00 は、『計測』完了からインターバル時間経過後に自動的に新たな『計測』を開始します。
- (9) 制御マイコンは、新たな『計測』が完了したことを確認するために、INTOUT 信号のアサートを確認します。制御マイコンは、INTOUT 信号がアサートしたのを確認後、Control 2 Register 【Address=0x40】へ 0x00 をライトして INTOUT 信号をネゲートし、Result Data Register 【Address=0x19】等のレジスタから計測結果を読み出します。
- (10) 以降、制御マイコンは、(9)を繰り返します。

以上に説明した制御手順に対応した制御フローを次ページに示します。

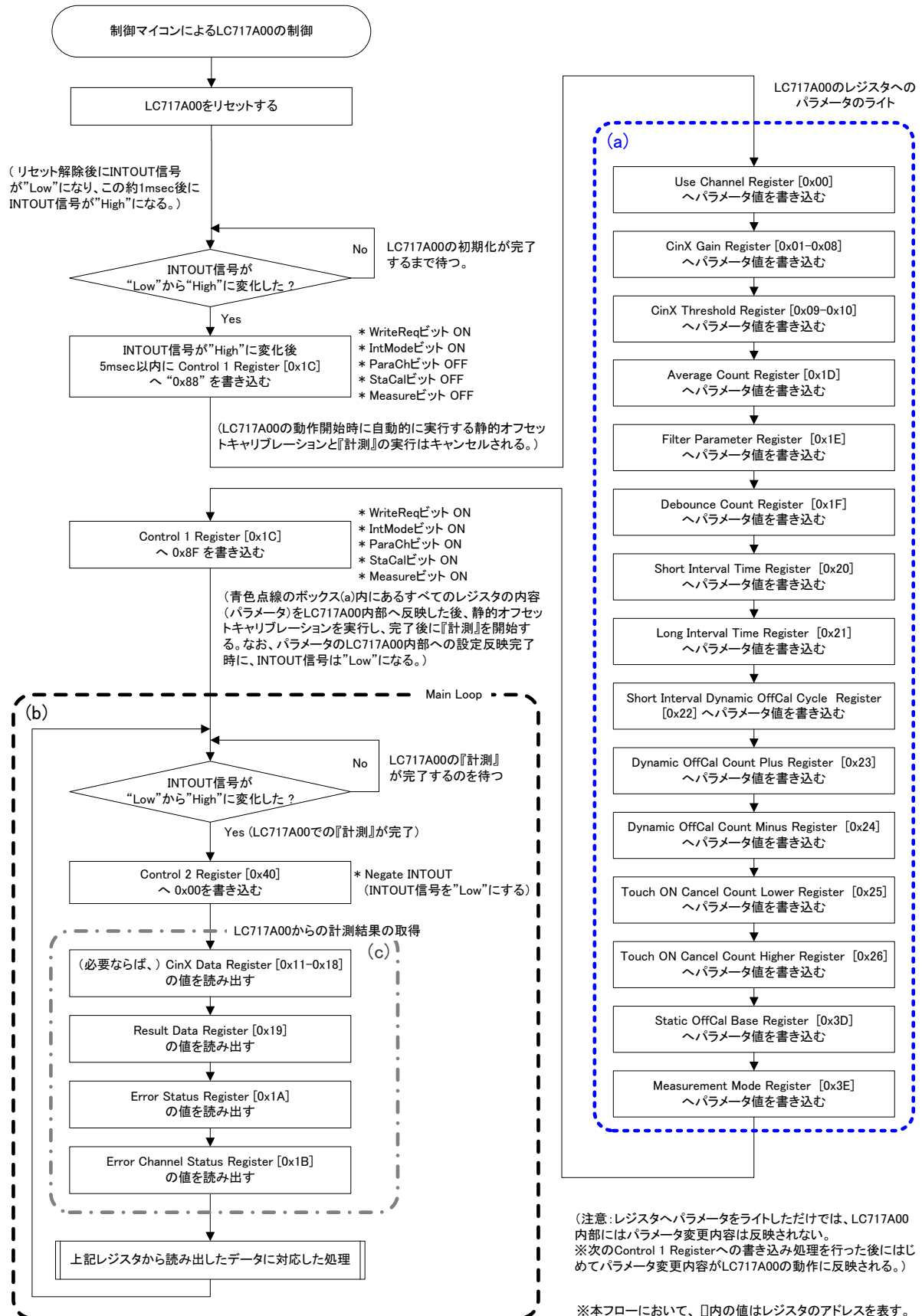


図 4-2 インターバルモードに設定し、INTOUT を使って計測結果をリードする場合の制御例

4.1.2 インターバルモードに設定し、INTOUTを使わずに計測結果をリードする場合

制御マイコンとLC717A00をI²C互換バス（またはSPIバス）で接続し、制御マイコンの別のGPIOポートとLC717A00のnRSTピンとを接続した場合を考えます。このような場合において、制御マイコンがINTOUT信号の状態をいっさい検出しないで50msec間隔でレジスタから計測結果を読み出す場合の制御例を説明します。

なお、本制御例では、LC717A00を常にインターバルモードに設定します。

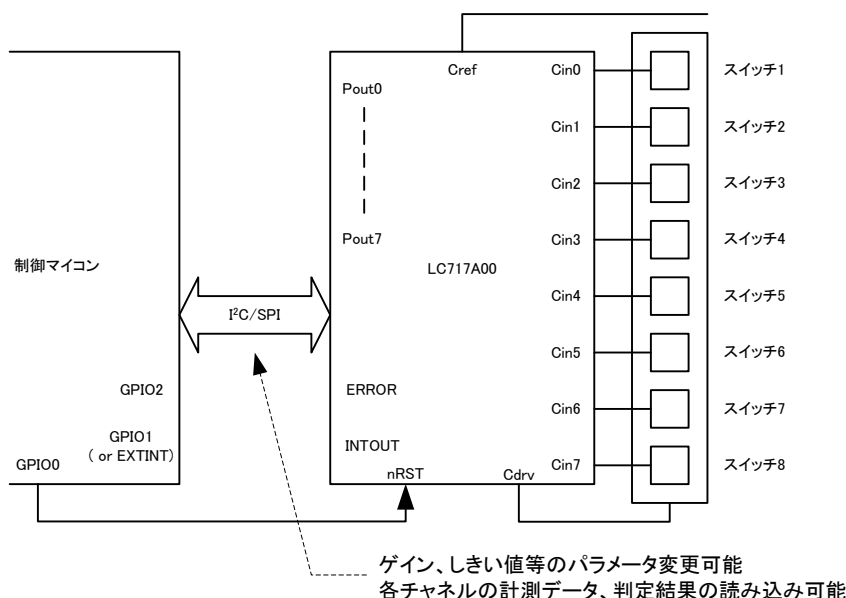


図 4-3 制御マイコンと LC717A00 との接続

- (1) 制御マイコンは、汎用出力ポートを使って nRST 信号を制御することにより LC717A00 をリセットします。
※具体的には、制御マイコンは、LC717A00 の nRST に接続した汎用出力ポートを使って nRST を” Low ” にした後再び” High ” にします。なお、最低 1 usec は” Low ” にします。
- (2) 制御マイコンは、リセット後に LC717A00 が自動実行する静的オフセットキャリブレーション処理が完了するまで待ちます。（例えば、nRST を使った LC717A00 のリセット制御終了直後から 100msec 経過するまで待ち、その後 Control 1 Register 【Address=0x1C】をリードして StaCal ビットが 0 になるまで待ちます。）
- (3) 制御マイコンは、新しい設定パラメータをレジスタ群にライトします。（具体的には、制御マイコンは、図 4-4 の青の点線で囲まれたボックス (A) 内のすべてのレジスタまたは一部のレジスタに新しいパラメータをライトします。）
※なお、この時点では、LC717A00 の内部にはまだ新しい設定パラメータの変更が反映されないことに注意してください。
- (4) 制御マイコンは、新しい設定を LC717A00 内部に反映させるとともに静的オフセットキャリブレーションと『計測』をスタートさせるために、Control 1 Register 【Address=0x1C】に 0x8F をライトします。

- (5) すると、LC717A00は、新しい設定をLC717A00内部に反映します。
- ※なお、新しい設定のLC717A00内部への反映完了時に、LC717A00のINTOUT信号は自動的にネゲートします（” Low” になります）。くわえて、LC717A00は、ERRORピンへの出力を” Low” に初期化します。また、出力ポートPout0～Pout7の出力信号をすべて” Low” に初期化します。また、LC717A00は、CinX Data Register 【Address=0x11～0x18】、Error Status Register 【Address=0x1A】、Error Channel Status Register 【Address=0x1B】のすべてのレジスタを0x00に初期化するとともに、すべてのチャネルのタッチON/OFF判定結果をOFFに初期化することに注意してください。くわえて、動的オフセットキャリブレーション処理用内部カウンタ、デバウンス処理用内部カウンタ、タッチON自動キャンセル処理用内部カウンタの値を0クリアすることに注意してください。
- (6) 次に、LC717A00は新しい設定で静的オフセットキャリブレーションを実行します。
- 制御マイコンは、LC717A00の静的オフセットキャリブレーション処理が完了したことを確認します。（例えば、Control 1 Register 【Address=0x1C】のStaCalビットが0になるまで待ちます。）
- (7) LC717A00は、静的オフセットキャリブレーション処理完了後、すぐに『計測』を開始します。
- (8) LC717A00は『計測』を完了します。『計測』完了からインターバル時間経過後にLC717A00は自動的に新たな『計測』を開始します。
- (9) 制御マイコンは、LC717A00の『計測』完了タイミングを考慮することなく、50msec毎にResult Data Register 【Address=0x19】等のレジスタから計測結果を読み出します。
- (10) 以降、LC717A00は(8)を繰り返し、制御マイコンは(9)を繰り返します。

以上に説明した制御手順に対応した制御フローを次ページに示します。

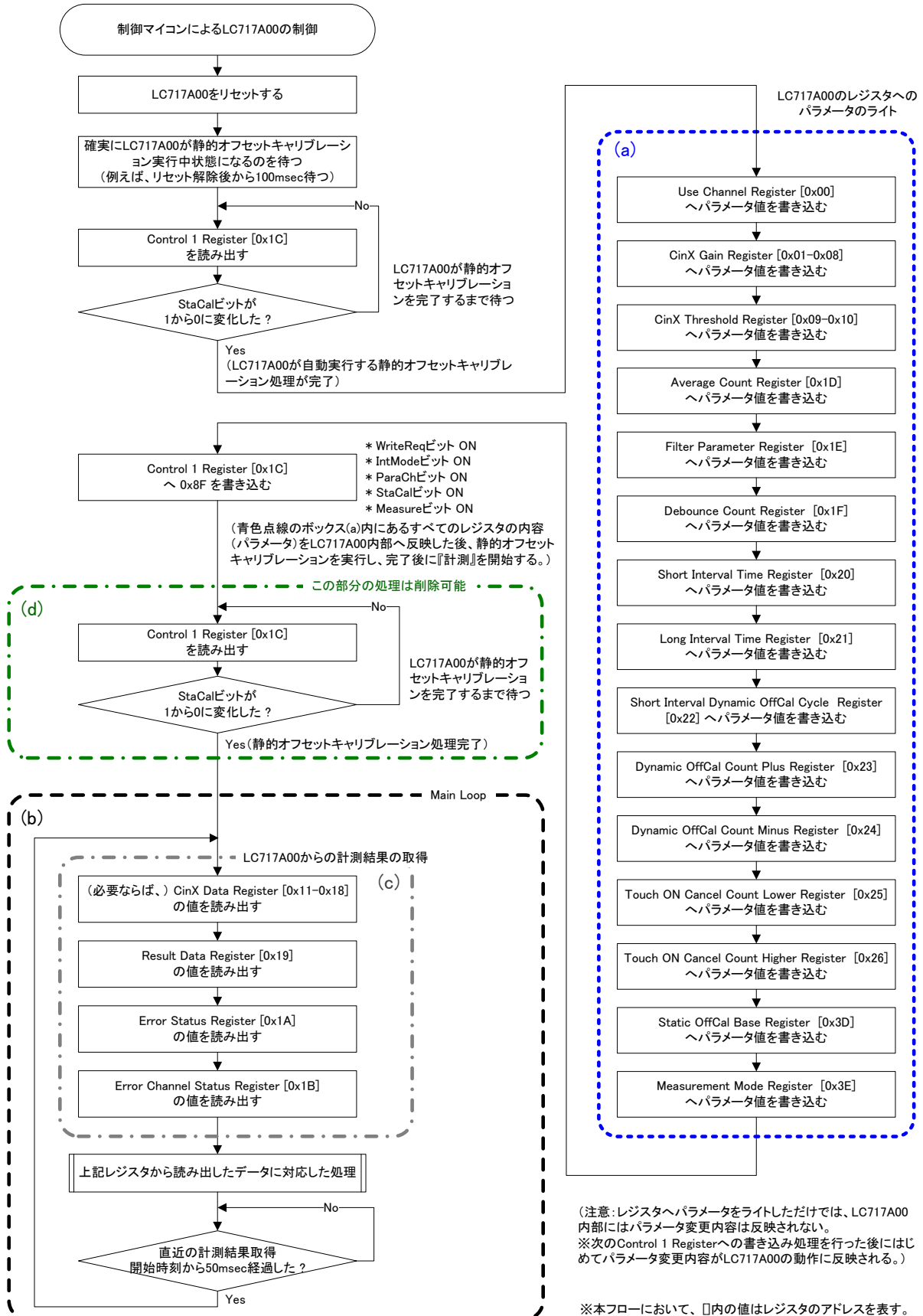


図 4-4 インターバルモードに設定し、INTOUT を使わずに計測結果をリードする場合の制御例

4.1.3 スリープモードに設定し、INTOUTを使って計測結果をリードする場合

制御マイコンとLC717A00をI²C互換バス（またはSPIバス）で接続し、制御マイコンのGPIOポートとLC717A00のINTOUTピンを接続し、制御マイコンの別のGPIOポートとLC717A00のnRSTピンを接続した場合を考えます。このような場合において、制御マイコンがINTOUT信号の状態を検出してレジスタから計測結果を読み出す場合の制御例を説明します。

なお、本制御例では、LC717A00を常にスリープモードに設定します。

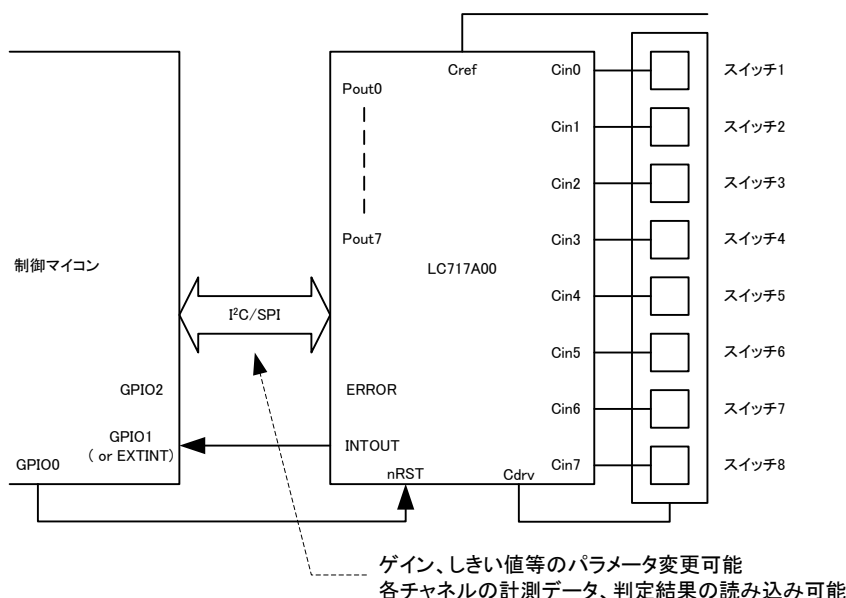


図 4-5 制御マイコンと LC717A00 との接続

- (1) 制御マイコンは、汎用出力ポートを使って nRST 信号を制御することにより LC717A00 をリセットします。
※具体的には、制御マイコンは、LC717A00 の nRST に接続した汎用出力ポートを使って nRST を "Low" にした後再び "High" にします。なお、最低 1 usec は "Low" にします。
- (2) 制御マイコンは、nRST を使った LC717A00 のリセット制御終了直後から汎用入力ポートを使って INTOUT 信号の状態を検出します。制御マイコンは、LC717A00 リセット解除後に発生する INTOUT 信号の立ち上がりエッジを検出したら、すみやかに Control 1 Register 【Address=0x1C】に 0x80 をライトします。（具体的には、LC717A00 リセット解除後に発生する INTOUT 信号の立ち上がりエッジから 5msec 以内に 0x80 をライトします。）これを行うのは、リセット後に LC717A00 が自動実行する静的オフセットキャリブレーションの実行と『計測』の実行を停止するためです。
- (3) LC717A00 はスリープ（動作停止）します。
制御マイコンは、新しい設定パラメータをレジスタ群にライトします。（具体的には、制御マイコンは、図 4-6 の青の点線で囲まれたボックス (A) 内のすべてのレジスタまたは一部のレジスタに新しいパラメータをライトします。）
※なお、この時点では、LC717A00 の内部にはまだ新しい設定パラメータの変更が反映されないことに注意してください。
- (4) 制御マイコンは、Control 1 Register 【Address=0x1C】に 0x87 をライトし、次に Control 2 Register 【Address=0x40】へ 0x01 をライトします。これにより、LC717A00 の INTOUT 信号はネゲートします（INTOUT 信号が "Low" になります）。これと同時に、LC717A00 は動作を再開します。

- (5) LC717A00は、「LC717A00のレジスタ群に設定した新しい設定パラメータのLC717A00内部への反映処理」を行います。

※なお、新しい設定のLC717A00内部への反映完了時に、LC717A00のINTOUT信号は自動的にネグートします(” Low” になります)。くわえて、LC717A00は、ERRORピンへの出力を” Low” に初期化します。また、出力ポートPout0～Pout7の出力信号をすべて” Low” に初期化します。

また、LC717A00は、CinX Data Register 【Address=0x11～0x18】、Error Status Register 【Address=0x1A】、Error Channel Status Register 【Address=0x1B】のすべてのレジスタを0x00に初期化するとともに、すべてのチャネルのタッチON/OFF判定結果をOFFに初期化することに注意してください。くわえて、動的オフセットキャリブレーション処理用内部カウンタ、デバウンス処理用内部カウンタ、タッチON自動キャンセル処理用内部カウンタの値を0クリアすることに注意してください。

- (6) 次に、LC717A00は新しい設定で静的オフセットキャリブレーションを実行します。そして、LC717A00はこの静的オフセットキャリブレーション完了後に『計測』を開始します。

- (7) LC717A00は、『計測』が完了すると、INTOUT信号をアサートした後(INTOUT信号が” High” になった後)にスリープします。

制御マイコンは、INTOUT信号がアサートしたことを確認した後にResult Data Register 【Address=0x19】等のレジスタから計測結果を読み出します。

- (8) 制御マイコンは、次の『計測』スタート予定時間まで待ちます。

- (9) 次の『計測』スタート予定時間になったら、制御マイコンはControl 2 Register 【Address=0x40】へ0x01をライトします。

- (10) LC717A00のINTOUT信号がネグートする(LC717A00のINTOUT信号が” Low” になる)とともに、LC717A00は動作を再開し、『計測』を開始します。

※なお、Control 2 Register 【Address=0x40】のWakeUpビットはLC717A00の動作再開時に自動的に0クリアされます。

- (11) 以降、(7)～(10)を繰り返します。

【特記事項】

スリープモードで動作しているLC717A00がスリープすると、

- ・制御マイコンがLC717A00をウェークアップしない限り、LC717A00は動作を再開しないこと
- ・今回の『計測』と次の『計測』の時間間隔は制御マイコンのウェークアップタイミングにのみ依存すること
- ・LC717A00がスリープモードで動作している時、LC717A00はShort Interval Time Register 【Address=0x20】およびLong Interval Time Register 【Address=0x21】の内容をいっさい参照しないこと

に注意してください。

以上に説明した制御手順に対応した制御フローを次ページに示します。

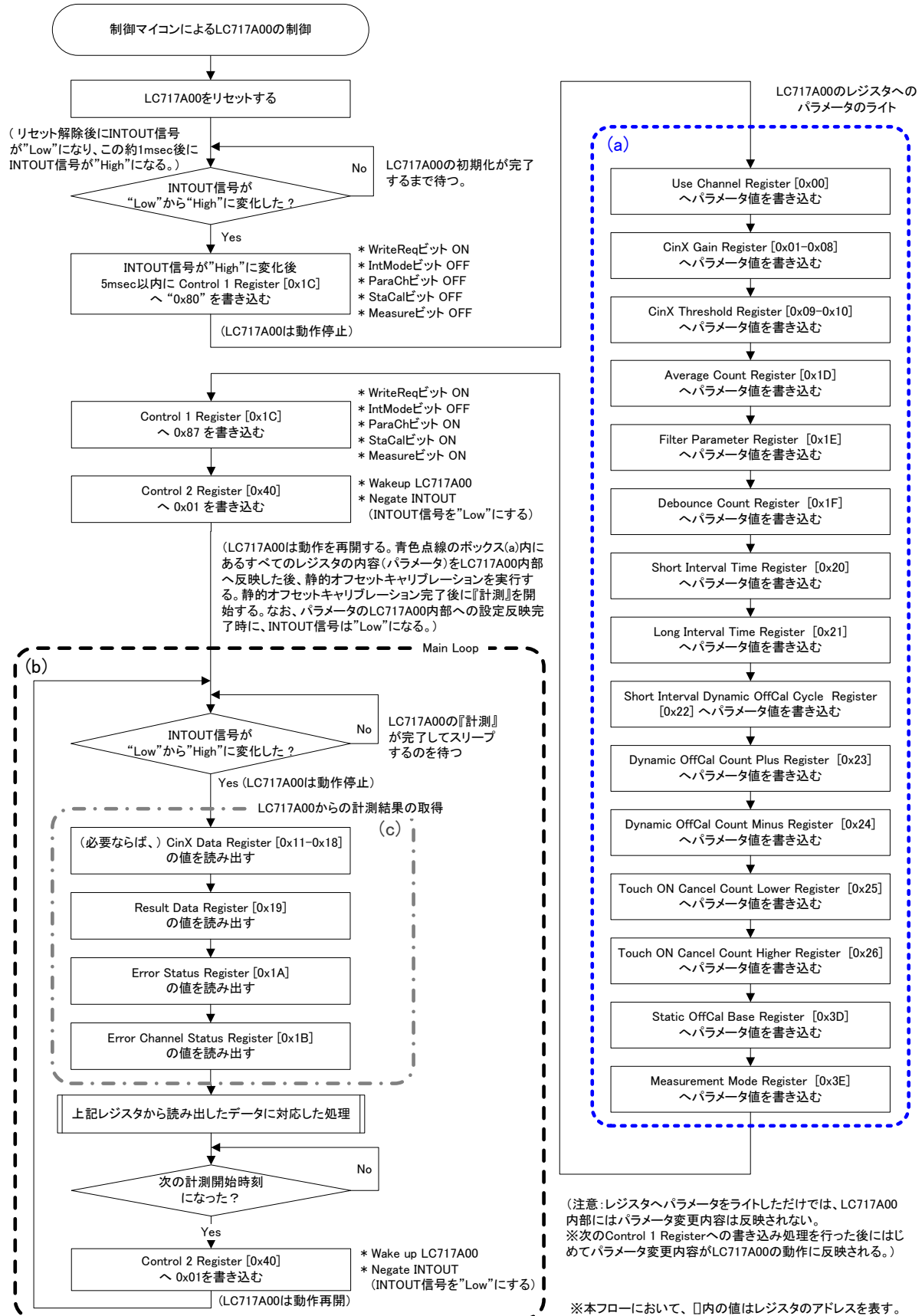


図 4-6 スリープモードに設定し、INTOUT を使って計測結果をリードする場合の制御例

4.1.4 インターバルモードに設定し、Pout0～7の出力をリードする場合

制御マイコンとLC717A00をI²C互換バス（またはSPIバス）で接続し、制御マイコンの8つのGPIOポートとLC717A00のPout0～Pout7を接続し、制御マイコンの別のGPIOポートとLC717A00のnRSTピンを接続した場合を考えます。このような場合において、制御マイコンがLC717A00からのレジスタリードとINTOUT信号状態の検出をいっさい行うことなく、50msec間隔で出力ポートPout0～Pout7の出力信号を制御マイコンのGPIOポートを介してリードする場合の制御例を説明します。

なお、本制御例では、制御マイコンは、LC717A00を常にインターバルモードに設定します。

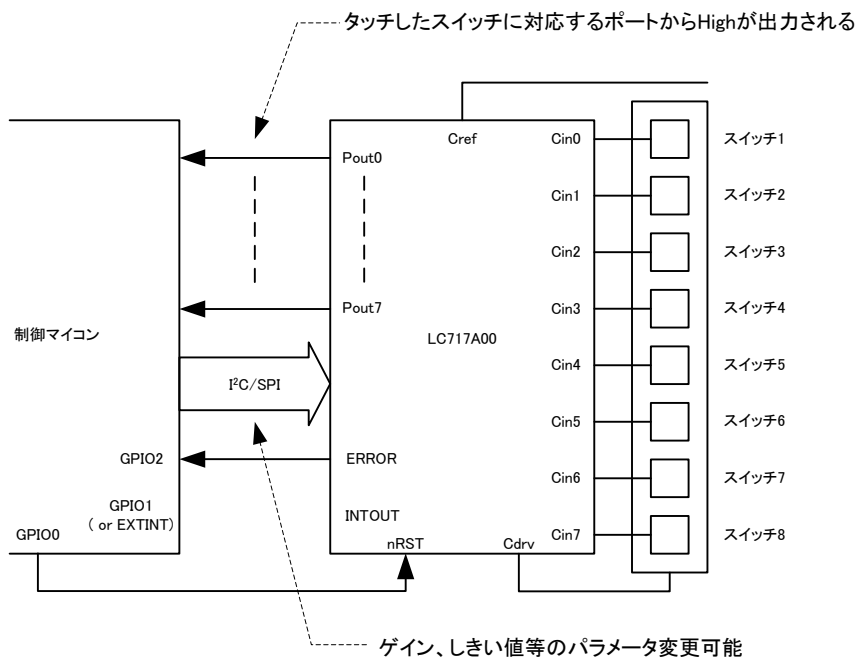


図 4-7 制御マイコンと LC717A00 との接続

- (1) 制御マイコンは、汎用出力ポートを使ってnRST信号を制御することによりLC717A00をリセットします。
※具体的には、制御マイコンは LC717A00 の nRST に接続した汎用出力ポートを使って nRST を” Low” にした後再び” High” にします。なお、最低 1 usec は” Low” にします。
- (2) LC717A00のリセット後、制御マイコンはLC717A00が静的オフセットキャリブレーション処理中の状態になるまで待ちます。（例えば、制御マイコンはnRSTを使ったLC717A00のリセット制御終了直後から100msec経過するまで待ちます。）
- (3) 制御マイコンはControl 1 Register 【Address=0x1C】に0x88をライトします。これにより、LC717A00は、現在行っている静的オフセットキャリブレーション処理が完了した後に『計測』を行わないようになります。
※なお、Control 1 Register 【Address=0x1C】に0x88をライトした時にすぐに静的オフセットキャリブレーション処理が中断しないことに注意してください。現在行われている静的オフセットキャリブレーション処理は最後まで実行されます。
- (4) 制御マイコンは、LC717A00で行われている静的オフセットキャリブレーション処理が確実に完了するまで待ちます。（例えば、制御マイコンは、LC717A00リセット解除後から500msec経過するまで待ちます。）
※なお、LC717A00が静的オフセットキャリブレーションしている間におけるLC717A00の出力ポートPout0～Pout7の出力信号はすべて” Low” となります。

- (5) 制御マイコンは、新しい設定パラメータをLC717A00のレジスタ群にライトします。(具体的には、制御マイコンは、図4-8の青の点線で囲まれたボックス(A)内のすべてのレジスタまたは一部のレジスタに新しいパラメータをライトします。)

※なお、この時点では、LC717A00の内部にはまだ新しい設定パラメータの変更が反映されないことに注意してください。

- (6) 次に、制御マイコンはControl 1 Register 【Address=0x1C】に0x8Fをライトします。

- (7) すると、LC717A00は、新しい設定をLC717A00内部に反映します。

※なお、新しい設定のLC717A00内部への反映完了時に、LC717A00のINTOUT信号は自動的にネグートします(“Low”になります)。くわえて、LC717A00は、ERRORピンへの出力を“Low”に初期化します。また、出力ポートPout0～Pout7の出力信号をすべて“Low”に初期化します。

また、LC717A00は、CinX Data Register 【Address=0x11～0x18】、Error Status Register 【Address=0x1A】、Error Channel Status Register 【Address=0x1B】のすべてのレジスタを0x00に初期化するとともに、すべてのチャネルのタッチON/OFF判定結果をOFFに初期化することに注意してください。くわえて、動的オフセットキャリブレーション処理用内部カウンタ、デバウンス処理用内部カウンタ、タッチON自動キャンセル処理用内部カウンタの値を0クリアすることに注意してください。

- (8) 次に、LC717A00は新しい設定で静的オフセットキャリブレーションを実行します。そして、LC717A00はこの静的オフセットキャリブレーション完了後に『計測』を開始します。

- (9) LC717A00は、『計測』を完了します。『計測』完了からインターバル時間経過後にLC717A00は自動的に新たな『計測』を開始します。

- (10) 以降、LC717A00は(9)を繰り返します。

一方、制御マイコンは、LC717A00の『計測』完了のタイミングをいっさい考慮しないで、50msec毎に制御マイコンのGPIOポートを介してLC717A00の出力ポートPout0～Pout7からの出力信号をリードします。

※なお、LC717A00においてキャリブレーションエラー等のエラーが発生すると、LC717A00のERRORピン出力が“High”になります。

以上に説明した制御手順に対応した制御フローを次ページに示します。

LC717A00 アプリケーションノート

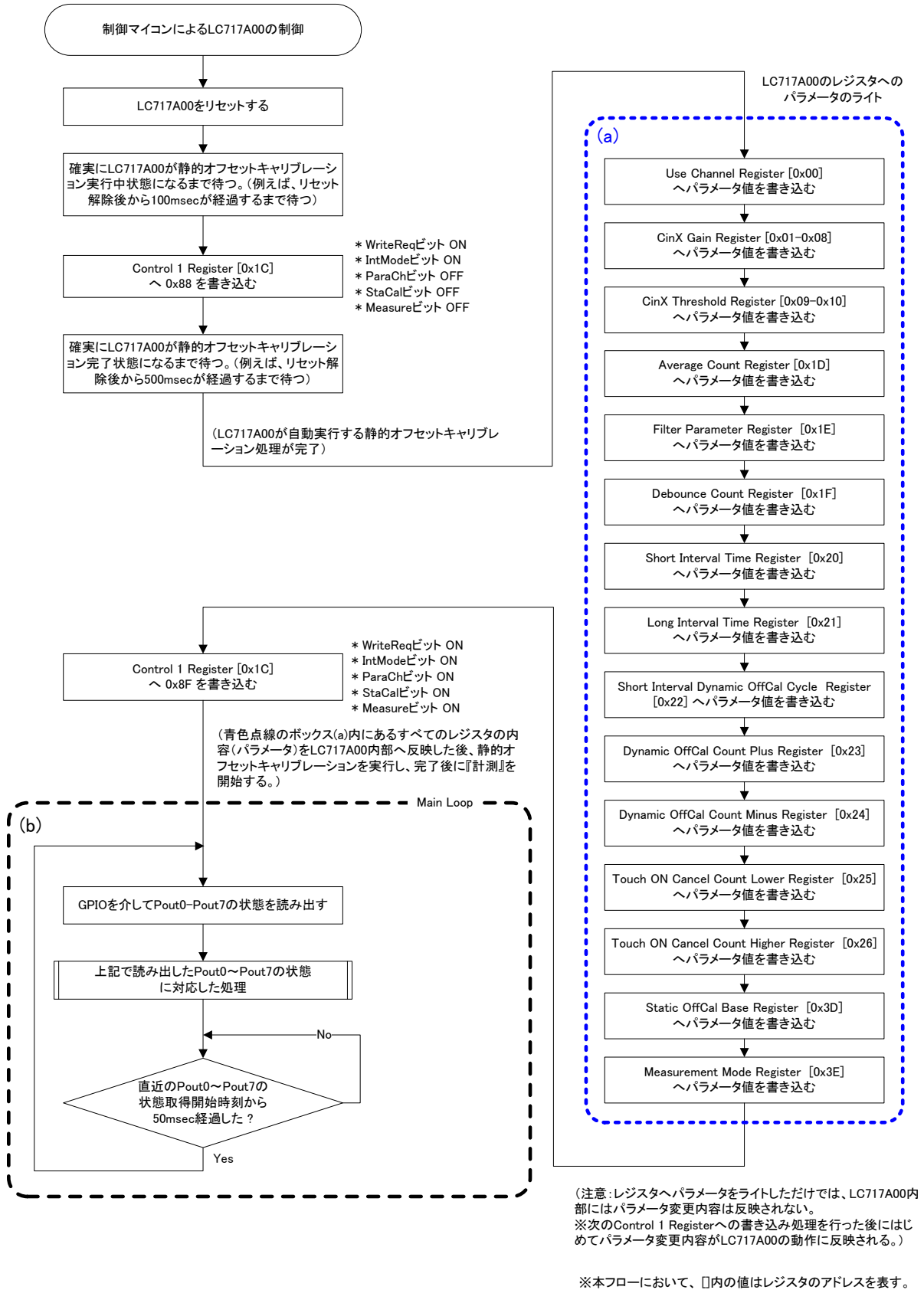


図 4-8 インターバルモード設定に設定し、Pout0~7 の出力をリードする場合の制御例

4.2 サンプルソース（擬似コード形式）

※免責事項

本ドキュメントに掲載したサンプル擬似コードは、お客様専用のソフトウェア開発の手助けのために作られたものです。お客様が必要と思う処理は適宜追加していただく必要があります。

本ドキュメントに掲載した擬似コードを参考にして作成したプログラムのご使用により万一お客様に損失や損害が生じたとしても、当社はその損失や損害について一切の責任を負いませんので、ご了承ください。

本節において、3つのサンプルソース（擬似コード）を示します。最初に、3つのサンプルソース内で使用する関数を以下に列挙します。

```
void I2c1byteWrite( unsigned char i2c_adr, unsigned char reg_adr, unsigned char data_w )
```

制御マイコンがI²Cスレーブアドレスi2c_adrをもつデバイスのレジスタアドレスreg_adrに1バイトデータdata_wをライトする関数です。

引数i2c_adrは、LC717A00のI²Cスレーブアドレスです。

引数reg_adrは、LC717A00のレジスタアドレスです。

引数data_wは、reg_adrで指定したアドレスのレジスタにライトする1バイトデータです。

```
unsigned char I2c1byteRead( unsigned char i2c_adr, unsigned char reg_adr )
```

制御マイコンがI²Cスレーブアドレスi2c_adrをもつデバイスのレジスタアドレスreg_adrから1バイトデータをリードする関数です。

引数i2c_adrは、LC717A00のI²Cスレーブアドレスです。

引数reg_adrは、LC717A00のレジスタアドレスです。

戻り値として、reg_adrで指定したアドレスのレジスタからリードした1バイトデータを返します。

```
void Wait_until_INTOUT_is_Low( void )
```

INTOUT 信号が” Low ” になるまで制御マイコンがウェイトする関数です。

```
void Wait_until_INTOUT_is_High( void )
```

INTOUT 信号が” High ” になるまで制御マイコンがウェイトする関数です。

```
void nRstCtrl_for_Reset( void )
```

LC717A00 をリセットするために、制御マイコンが nRST に” Low ” を出力した後、” High ” を出力する関数です。（nRST への出力を1 usec 以上の間”Low”に保つ必要があります。）

```
unsigned short GetNowTime_msec( void )
```

制御マイコンが1ミリ秒単位で現在時刻を得るための関数です。

戻り値として、2バイトの現在時刻を返します。

```
void Wait_for_100msec( void )
```

制御マイコンが100ミリ秒ウェイトするための関数です。

```
void Wait_for_50msec( void )
```

制御マイコンが50ミリ秒ウェイトするための関数です。

※実際にお使いになるシステム向けにこれらの関数を実装する必要があります。

LC717A00 アプリケーションノート

次に、3 つのサンプルソース内で共通で使用するヘッダファイル（ファイル名：“LC717A00.h”）の内容を以下に示します。

```
//-----  
//                               LC717A00 用インクルードファイル  
//                               (LC717A00.h)  
//-----  
  
#define SLAVE_ADDRESS          (0x16)    // LC717A00のI2Cスレーブアドレス  
//                               (0x16 or 0x17)  
  
#define USECHANNEL_REG         (0x00)    // Use Channel Registerのアドレス  
#define CIN0GAIN_REG           (0x01)    // Cin0 Gain Registerのアドレス  
#define CIN1GAIN_REG           (0x02)    // Cin1 Gain Registerのアドレス  
#define CIN2GAIN_REG           (0x03)    // Cin2 Gain Registerのアドレス  
#define CIN3GAIN_REG           (0x04)    // Cin3 Gain Registerのアドレス  
#define CIN4GAIN_REG           (0x05)    // Cin4 Gain Registerのアドレス  
#define CIN5GAIN_REG           (0x06)    // Cin5 Gain Registerのアドレス  
#define CIN6GAIN_REG           (0x07)    // Cin6 Gain Registerのアドレス  
#define CIN7GAIN_REG           (0x08)    // Cin7 Gain Registerのアドレス  
#define CIN0THRESHOLD_REG      (0x09)    // Cin0 Threshold Registerのアドレス  
#define CIN1THRESHOLD_REG      (0x0A)    // Cin1 Threshold Registerのアドレス  
#define CIN2THRESHOLD_REG      (0x0B)    // Cin2 Threshold Registerのアドレス  
#define CIN3THRESHOLD_REG      (0x0C)    // Cin3 Threshold Registerのアドレス  
#define CIN4THRESHOLD_REG      (0x0D)    // Cin4 Threshold Registerのアドレス  
#define CIN5THRESHOLD_REG      (0x0E)    // Cin5 Threshold Registerのアドレス  
#define CIN6THRESHOLD_REG      (0x0F)    // Cin6 Threshold Registerのアドレス  
#define CIN7THRESHOLD_REG      (0x10)    // Cin7 Threshold Registerのアドレス  
#define CIN0DATA_REG           (0x11)    // Cin0 Data Registerのアドレス  
#define CIN1DATA_REG           (0x12)    // Cin1 Data Registerのアドレス  
#define CIN2DATA_REG           (0x13)    // Cin2 Data Registerのアドレス  
#define CIN3DATA_REG           (0x14)    // Cin3 Data Registerのアドレス  
#define CIN4DATA_REG           (0x15)    // Cin4 Data Registerのアドレス  
#define CIN5DATA_REG           (0x16)    // Cin5 Data Registerのアドレス  
#define CIN6DATA_REG           (0x17)    // Cin6 Data Registerのアドレス  
#define CIN7DATA_REG           (0x18)    // Cin7 Data Registerのアドレス  
#define RSLTDATA_REG           (0x19)    // Result Data Registerのアドレス  
#define ERRSTS_REG             (0x1A)    // Error Status Registerのアドレス  
#define ERRCHSTS_REG           (0x1B)    // Error Channel Status Registerのアドレス  
#define CONTROL1_REG           (0x1C)    // Control 1 Registerのアドレス  
#define AVECNT_REG             (0x1D)    // Average Count Registerのアドレス  
#define FILPRM_REG             (0x1E)    // Filter Parameter Registerのアドレス  
#define DEBCNT_REG             (0x1F)    // Debounce Count Registerのアドレス  
#define SHORTITVL_REG          (0x20)    // Short Interval Time Registerのアドレス  
#define LONGITVL_REG           (0x21)    // Long Interval Time Registerのアドレス  
#define SIDCALCYCLE_REG         (0x22)    // Short Interval Dynamic OffCal Cycle Register  
//                               のアドレス  
#define DCALCNTPLUS_REG         (0x23)    // Dynamic OffCal Count Plus Registerのアドレス  
#define DCALCNTMINUS_REG        (0x24)    // Dynamic OffCal Count Minus Registerのアドレス  
#define TCHONCNCLCNTL_REG       (0x25)    // Touch ON Cancel Count Lower Registerのアドレス
```

LC717A00 アプリケーションノート

```
#define TCHONCNCLCNTH_REG    (0x26)    // Touch ON Cancel Count Higher Register
                                // のアドレス
#define SCALCDACBASE_REG     (0x3D)    // Static OffCal Base Registerのアドレス
#define MEASMODE_REG         (0x3E)    // Measurement Mode Registerのアドレス
#define CONTROL2_REG         (0x40)    // Control 2 Registerのアドレス

// CONTROL 1 Register 向け
#define WRITE_REQ_BIT         (0x80)    // WriteReqビット
#define INT_MODE_BIT          (0x08)    // IntModeビット
#define PARA_CH_BIT           (0x04)    // ParaChビット
#define STA_CAL_BIT           (0x02)    // StaCalビット
#define MEASURE_BIT           (0x01)    // Measureビット

// CONTROL 2 Register 向け
#define WAKEUP_BIT            (0x01)    // WakeUpビット
```

4.2.1 インターバルモードに設定し、INTOUTを使って計測結果をリードする場合

LC717A00と制御マイコンをI²C互換バスで接続した場合において、制御マイコンがINTOUT信号の状態をチェックしてレジスタから計測結果を読み出す場合のサンプルソース（擬似コード）を示します。

なお、このサンプルソースでは、制御マイコンは、LC717A00の動作モードを常にインターバルモードに設定しています。

※一般的に、LC717A00はスリープモードではなくインターバルモードで動作させます。

```
//-----  
//          LC717A00 用サンプルプログラム  
//          (制御マイコンがLC717A00のINTOUT信号を使用する場合)  
//-----  
  
// インクルードファイル  
#include "LC717A00.h"  
  
// 関数プロトタイプ宣言  
void nRstCtrl_for_Reset( void );  
void Wait_until_INTOUT_is_Low( void );  
void Wait_until_INTOUT_is_High( void );  
void I2c1byteWrite( unsigned char, unsigned char, unsigned char );  
unsigned char I2c1byteRead( unsigned char, unsigned char );  
  
void main( void )  
{  
    unsigned char w_data, cnt, rslt_data;  
    unsigned char err_sts, errch_sts;  
  
    /*****  
    /**   LC717A00のリセット&リセット解除を行います。      **/  
    /**   ****/  
    // (制御マイコンは、LC717A00のnRSTに接続した汎用出力ポートを使ってnRSTを” Low” に  
    //   した後再び” High” にします。なお、nRSTへの出力を1 usec以上の間”Low”にします。)  
    nRstCtrl_for_Reset();  
  
    /*****  
    /**   制御マイコンは、LC717A00のリセットが解除されたこと、      **/  
    /**   および LC717A00内部の初期化が完了したことを確認します。  **/  
    /**   ****/  
    //制御マイコンは、LC717A00においてリセットが解除されたことを確認します。  
    // (具体的には、制御マイコンは、LC717A00のINTOUTに接続した汎用入力ポートを使って  
    // INTOUTが” Low” になったことを確認します。)  
    Wait_until_INTOUT_is_Low();          // INTOUTが” Low” になるまで待ちます。  
  
    // 【注意】 インタフェースとしてSPIを選択した時の注意
```

```
// インタフェースとしてSPIを選択した時は、
// このタイミング（INTOUTが” Low” になったことを検出したタイミング）で、
// nCSピンへの出力を最低 1 usec以上の間” Low” にしてください。
//
// nCS = 1;
// nCS = 0;
// nCS = 1;
//

//制御マイコンは、LC717A00内部の初期化が完了したことを確認します。
//（具体的には、制御マイコンは、LC717A00のINTOUTに接続した汎用入力ポートを使って
//リセット解除後にINTOUTが” Low” から” High” に変化したのを検出するまで待ちます。）
Wait_until_INTOUT_is_High(); // INTOUTが” High” になるまで待ちます。

/*****
**      LC717A00 で自動実行される予定であった静的オフセット      **/
**      キャリブレーションや『計測』の実行をキャンセルします。    **/
*****/
//INTOUTが” Low” から” High” になった後
//すみやかに（目安として5msec以内に）Control 1 Registerに0x88をライトします。
w_data = (WRITE_REQ_BIT | INT_MODE_BIT);
I2c1byteWrite( SLAVE_ADDRESS, CONTROL1_REG, w_data );

/*****
**      パラメータ設定処理の一例です。                          **/
*****/
// 1. 使用するチャネルをCin0-Cin5に設定します。（設定値は0x3F）
// 2. Cin0-Cin5のチャネルにおいて、初段アンプゲインを1600[ff]、
//    次段アンプゲインを2倍に設定します。（設定値は0x10）
// 3. Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
// 4. ノイズ対策用パラメータを0x0Cに設定します。
//※他のレジスタの設定もこれらと同様に行います。
//※必要に応じて、パラメータ設定処理を行った後に、
// パラメータを設定したすべてのレジスタの値をリードすることにより、
// 設定値のバリファイを行います。

//使用するチャネルをCin0-Cin5に設定します。（設定値は0x3F）
I2c1byteWrite( SLAVE_ADDRESS, USECHANNEL_REG, 0x3F );

//Cin0-Cin5のチャネルにおいて、初段アンプゲインを1600[ff]、
//次段アンプゲインを2倍に設定します。（設定値は0x10）
w_data = 0x10;
for( cnt=0; cnt<=5; cnt++ )
{
    I2c1byteWrite( SLAVE_ADDRESS, (CINOGAIN_REG+cnt), w_data );
}

//Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
w_data = 0x0A;
for( cnt=0; cnt<=5; cnt++ )
```

```
{
    I2c1byteWrite( SLAVE_ADDRESS, (CIN0THRESHOLD_REG+cnt), w_data );
}

//ノイズ対策用パラメータを0x0Cに設定します。
I2c1byteWrite( SLAVE_ADDRESS, FILPRM_REG, 0x0C );

/*****/
/**   インターバルモード + パラメータ設定要求 +           **/
/**   静的オフセットキャリブレーション要求 + 『計測』 を設定します。   **/
/*****/
w_data = (WRITE_REQ_BIT | INT_MODE_BIT | PARA_CH_BIT | STA_CAL_BIT | MEASURE_BIT);
I2c1byteWrite( SLAVE_ADDRESS, CONTROL1_REG, w_data );

/*****/
/**   INTOUT信号の状態をチェックしながら           **/
/**   タッチ検出結果を読み出すためのコード例です。   **/
/*****/
for(;;)
{
    //INTOUT信号が” High” になるまで待ちます。
    // (LC717A00において『計測』が終了するまで待ちます。)
    Wait_until_INTOUT_is_High();

    //Control 2 Registerへ0x00をライトします。
    I2c1byteWrite( SLAVE_ADDRESS, CONTROL2_REG, 0x00 );

    /*****/
    /**   タッチ検出結果を読み出します。   **/
    /*****/
    //必要に応じて、CinX Data Registerをリードします。

    //Result Data Registerをリードします。
    rslt_data = I2c1byteRead( SLAVE_ADDRESS, RSLTDATA_REG );

    //Error Status Registerをリードします。
    err_sts = I2c1byteRead( SLAVE_ADDRESS, ERRSTS_REG );

    //Error Channel Status Registerをリードします。
    errch_sts = I2c1byteRead( SLAVE_ADDRESS, ERRCHSTS_REG );

    //読み出したタッチ検出結果に対応した処理を記述します。
    //          :
    //          :
    //          :
```

```
//      :  
//      :  
}  
  
}
```

4.2.2 インターバルモードに設定し、INTOUTを使わずに計測結果をリードする場合

LC717A00と制御マイコンをI²C互換バスで接続した場合において、制御マイコンがINTOUT信号の状態をいっさいチェックすることなく50msec間隔でレジスタから計測結果を読み出す場合のサンプルソース（擬似コード）を示します。

なお、このサンプルソースでは、制御マイコンは、LC717A00の動作モードを常にインターバルモードに設定しています。

※一般的に、LC717A00はスリープモードではなくインターバルモードで動作させます。

```
//-----  
//          LC717A00 用サンプルプログラム  
//          (制御マイコンがLC717A00のINTOUT信号を使用しない場合)  
//-----  
  
// インクルードファイル  
#include "LC717A00.h"  
  
// 関数プロトタイプ宣言  
void nRstCtrl_for_Reset( void );  
void Wait_for_100msec( void );  
void I2c1byteWrite( unsigned char, unsigned char, unsigned char );  
unsigned char I2c1byteRead( unsigned char, unsigned char );  
unsigned short GetNowTime_msec( void );  
  
void main( void )  
{   unsigned char r_data, w_data, cnt, rslt_data;  
    unsigned char err_sts, errch_sts;  
    unsigned short sta_time_msec, now_time_msec;  
  
    /*****  
    /**   LC717A00のリセット&リセット解除を行います。      **/  
    *****/  
    // (制御マイコンは、LC717A00のnRSTに接続した汎用出力ポートを使ってnRSTを” Low” に  
    //   した後再び” High” にします。なお、nRSTへの出力を1 usec以上の間”Low”にします。)  
    nRstCtrl_for_Reset();  
  
    /*****  
    /**   制御マイコンは、LC717A00の静的オフセットキャリ      **/  
    /**   ブレーションが終了するまで待ちます。              **/  
    *****/  
    //制御マイコンは、LC717A00が静的オフセットキャリブレーション実行中の  
    //状態になるまで待ちます。（例えば、100msec待ちます）。  
    Wait_for_100msec();  
  
    // 【注意】インタフェースとしてSPIを選択した時の注意  
    // インタフェースとしてSPIを選択した時は、
```

```
// LSI内部の初期化が確実に終了したタイミング（例えば、リセットを解除してから50msec
// 後のタイミング）で、nCSピンへの出力を最低1 usec以上の間” Low” にしてください。
// （具体的には、nCSに” High” →” Low” →” High” を出力。）
//
// Wait_for_50msec();
// nCS = 1;
// nCS = 0;
// nCS = 1;
// Wait_for_50msec();
//
```

//制御マイコンは、静的オフセットキャリブレーションが終了するまで待ちます。

```
for(;;)
{
    r_data = I2c1byteRead( SLAVE_ADDRESS, CONTROL1_REG );

    //WriteReqビット, ParaChビット, StaCalビットがすべて0になったことを確認します。
    if( ( r_data & ( WRITE_REQ_BIT | PARA_CH_BIT | STA_CAL_BIT ) ) == 0x00 )
    {
        break;
    }
}
```

```
/******
**                パラメータ設定処理の一例です。                **
/******
// 1. 使用するチャンネルをCin0-Cin5に設定します。（設定値は0x3F）
// 2. Cin0-Cin5のチャンネルにおいて、初段アンプゲインを1600[fF]、
//    次段アンプゲインを2倍に設定します。（設定値は0x10）
// 3. Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
// 4. ノイズ対策用パラメータを0x0Cに設定します。
//※他のレジスタの設定もこれらと同様に行います。
//※必要に応じて、パラメータ設定処理を行った後に、
// パラメータを設定したすべてのレジスタの値をリードすることにより、
// 設定値のベリファイを行います。
```

```
//使用するチャンネルをCin0-Cin5に設定します。（設定値は0x3F）
I2c1byteWrite( SLAVE_ADDRESS, USECHANNEL_REG, 0x3F );
```

```
//Cin0-Cin5のチャンネルにおいて、初段アンプゲインを1600[fF]、
//次段アンプゲインを2倍に設定します。（設定値は0x10）
```

```
w_data = 0x10;
for( cnt=0; cnt<=5; cnt++ )
{
    I2c1byteWrite( SLAVE_ADDRESS, (CINOGAIN_REG+cnt), w_data );
}
```

```
//Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
w_data = 0x0A;
for( cnt=0; cnt<=5; cnt++ )
```



```
{
    I2c1byteWrite( SLAVE_ADDRESS, (CINOTHRESHOLD_REG+cnt), w_data );
}

//ノイズ対策用パラメータを0x0Cに設定します。
I2c1byteWrite( SLAVE_ADDRESS, FILPRM_REG, 0x0C );

/*****
/**   インターバルモード + パラメータ設定要求 +
/**   静的オフセットキャリブレーション要求 + 『計測』 を設定します。
*****/
w_data = (WRITE_REQ_BIT | INT_MODE_BIT | PARA_CH_BIT | STA_CAL_BIT | MEASURE_BIT);
I2c1byteWrite( SLAVE_ADDRESS, CONTROL1_REG, w_data );

/*****
/**   静的オフセットキャリブレーションが完了するまで待ちます。
/**   (なお、この待ち処理のコードは、削除しても問題なく動作します。)
*****/
for(;;)
{
    r_data = I2c1byteRead( SLAVE_ADDRESS, CONTROL1_REG );

    //WriteReqビット, ParaChビット, StaCalビットがすべて0になったことを確認します。
    if( ( r_data & ( WRITE_REQ_BIT | PARA_CH_BIT | STA_CAL_BIT ) ) == 0x00 )
    {
        break;
    }
}

/*****
/**   INTOUT信号の状態をチェックすることなく、
/**   タッチ検出結果を読み出すためのコード です。
*****/
//現在時刻をmsec単位で取得して保存します。
sta_time_msec = GetNowTime_msec();
for(;;)
{
    /*****
    /**   タッチ検出結果を読み出します。
    *****/

    //必要に応じて、CinX Data Registerをリードします。

    //Result Data Registerをリードします。
    rslt_data = I2c1byteRead( SLAVE_ADDRESS, RSLTDATA_REG );

    //Error Status Registerをリードします。
    err_sts = I2c1byteRead( SLAVE_ADDRESS, ERRSTS_REG );
```

```
//Error Channel Status Registerをリードします。  
errch_sts = I2c1byteRead( SLAVE_ADDRESS, ERRCHSTS_REG );
```

```
//読み出したタッチ検出結果に対応した処理を記述します。
```

```
//      :  
//      :  
//      :  
//      :  
//      :  
/*****/  
/** 直近の計測結果取得開始時刻から50msec経過するまで待ちます。  **/  
/*****/  
for(;;)  
{  
    //現在時刻をmsec単位で取得して保存します。  
    now_time_msec = GetNowTime_msec();  
    if( (now_time_msec - sta_time_msec)>=50 )  
    {  
        break;  
    }  
}  
sta_time_msec = now_time_msec;  
}  
  
}
```

4.2.3 スリープモードに設定し、INTOUTを使って計測結果をリードする場合

LC717A00と制御マイコンをI²C互換バスで接続した場合において、制御マイコンがINTOUT信号の状態をチェックしてレジスタから計測結果を読み出す場合のサンプルソース（擬似コード）を示します。

なお、このサンプルソースでは、制御マイコンは、LC717A00の動作モードを常にスリープモードに設定しています。

```
//-----  
//          LC717A00 用サンプルプログラム  
//          (制御マイコンがLC717A00をスリープモードに設定する場合)  
//-----  
  
// インクルードファイル  
#include "LC717A00.h"  
  
// 関数プロトタイプ宣言  
void nRstCtrl_for_Reset( void );  
void Wait_until_INTOUT_is_Low( void );  
void Wait_until_INTOUT_is_High( void );  
void I2c1byteWrite( unsigned char, unsigned char, unsigned char );  
unsigned char I2c1byteRead( unsigned char, unsigned char );  
  
void main( void )  
{  
    unsigned char w_data, cnt, rslt_data;  
    unsigned char err_sts, errch_sts;  
    unsigned short sta_time_msec, now_time_msec;  
  
    /*-----*/  
    /**    LC717A00のリセット&リセット解除を行います。    **/  
    /*-----*/  
    // (制御マイコンは、LC717A00のnRSTに接続した汎用出力ポートを使ってnRSTを” Low” に  
    //   した後再び” High” にします。なお、nRSTへの出力を1 usec以上の間”Low”にします。)  
    nRstCtrl_for_Reset();  
  
    /*-----*/  
    /**    制御マイコンは、LC717A00のリセットが解除されたこと、    **/  
    /**    および LC717A00内部の初期化が完了したことを確認します。    **/  
    /*-----*/  
    //制御マイコンは、LC717A00においてリセットが解除されたことを確認します。  
    // (具体的には、制御マイコンは、LC717A00のINTOUTに接続した汎用入力ポートを使って  
    // INTOUTが” Low” になったことを確認します。)  
    Wait_until_INTOUT_is_Low();          // INTOUTが” Low” になるまで待ちます。  
  
    // 【注意】 インタフェースとしてSPIを選択した時の注意
```

```
// インタフェースとしてSPIを選択した時は、
// このタイミング（INTOUTが” Low” になったことを検出したタイミング）で、
// nCSピンへの出力を最低1 usec以上の間” Low” にしてください。
// （具体的には、nCSに” High” →” Low” →” High” を出力。）
//
// nCS = 1;
// nCS = 0;
// nCS = 1;
//

//制御マイコンは、LC717A00内部の初期化が完了したことを確認します。
//（具体的には、制御マイコンは、LC717A00のINTOUTに接続した汎用入力ポートを使って
//リセット解除後にINTOUTが” Low” から” High” に変化したのを検出するまで待ちます。）
Wait_until_INTOUT_is_High(); // INTOUTが” High” になるまで待ちます。

/*****
**      LC717A00 で自動実行される予定であった静的オフセット      **/
**      キャリブレーションや『計測』の実行をキャンセルすると      **/
**      ともに、LC717A00をスリープさせます。                      **/
*****/
//INTOUTが” Low” から” High” になった後
//すみやかに（目安として5msec以内に）Control 1 Registerに0x80をライトします。
w_data = WRITE_REQ_BIT;
I2c1byteWrite( SLAVE_ADDRESS, CONTROL1_REG, w_data );

/*****
**      パラメータ設定処理の一例です。                          **/
*****/
// 1．使用するチャネルをCin0-Cin5に設定します。（設定値は0x3F）
// 2．Cin0-Cin5のチャネルにおいて、初段アンプゲインを1600[fF]、
//   次段アンプゲインを2倍に設定します。（設定値は0x10）
// 3．Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
// 4．ノイズ対策用パラメータを0x0Cに設定します。
//※他のレジスタの設定もこれらと同様に行います。
//※必要に応じて、パラメータ設定処理を行った後に、
// パラメータを設定したすべてのレジスタの値をリードすることにより、
// 設定値のベリファイを行います。

//使用するチャネルをCin0-Cin5に設定します。（設定値は0x3F）
I2c1byteWrite( SLAVE_ADDRESS, USECHANNEL_REG, 0x3F );

//Cin0-Cin5のチャネルにおいて、初段アンプゲインを1600[fF]、
//次段アンプゲインを2倍に設定します。（設定値は0x10）
w_data = 0x10;
for( cnt=0; cnt<=5; cnt++ )
{
    I2c1byteWrite( SLAVE_ADDRESS, (CIN0GAIN_REG+cnt), w_data );
}

//Cin0-Cin5のしきい値レベルを10に設定します。（設定値は0x0A）
```

```
w_data = 0x0A;
for( cnt=0; cnt<5; cnt++ )
{
    I2c1byteWrite( SLAVE_ADDRESS, (CIN0THRESHOLD_REG+cnt), w_data );
}

//ノイズ対策用パラメータを0x0Cに設定します。
I2c1byteWrite( SLAVE_ADDRESS, FILPRM_REG, 0x0C );

/*****
/**   スリープモード + パラメータ設定要求 +                               **/
/**   静的オフセットキャリブレーション要求 + 『計測』 を設定します。       **/
*****/
w_data = (WRITE_REQ_BIT | PARA_CH_BIT | STA_CAL_BIT | MEASURE_BIT);
I2c1byteWrite( SLAVE_ADDRESS, CONTROL1_REG, w_data );

/*****
/**   LC717A00をウェークアップします。   **/
*****/
I2c1byteWrite( SLAVE_ADDRESS, CONTROL2_REG, WAKEUP_BIT );

//変数sta_time_msecにダミーの初期値を格納します。
sta_time_msec = GetNowTime_msec();

/*****
/**   制御マイコンが約50msec間隔でLC717A00をウェークアップ   **/
/**   してタッチ検出結果を読み出すためのコード例です。       **/
*****/
for(;;)
{
    //INTOUT信号が” High” になるまで待ちます。
    // (LC717A00において『計測』が終了してスリープするまで待ちます。)
    Wait_until_INTOUT_is_High();

    /*****
    /**   タッチ検出結果を読み出します。   **/
    *****/
    //必要に応じて、CinX Data Registerをリードします。

    //Result Data Registerをリードします。
    rslt_data = I2c1byteRead( SLAVE_ADDRESS, RSLTDATA_REG );

    //Error Status Registerをリードします。
```

```
err_sts = I2c1byteRead( SLAVE_ADDRESS, ERRSTS_REG );

//Error Channel Status Registerをリードします。
errch_sts = I2c1byteRead( SLAVE_ADDRESS, ERRCHSTS_REG );

//読み出したタッチ検出結果に対応した処理を記述します。
//      :
//      :
//      :
//      :
//      :

/*****/
/** 直近の計測結果取得開始時刻から50msec経過するまで待ちます。  **/
/*****/
for(;;)
{
    //現在時刻をmsec単位で取得して保存します。
    now_time_msec = GetNowTime_msec();
    if( (now_time_msec - sta_time_msec)>=50 )
    {
        break;
    }
}
sta_time_msec = now_time_msec;

/*****/
/** INTOUT信号をネゲートするとともに  **/
/** LC717A00をウェークアップします。  **/
/*****/
I2c1byteWrite( SLAVE_ADDRESS, CONTROL2_REG, WAKEUP_BIT );

}

}
```

LC717A00 アプリケーションノート

* : I²C バスはフィリップス社の商標です。

ON Semiconductor及びONのロゴはSemiconductor Components Industries, LLC (SCILLC)の登録商標です。SCILLCは特許、商標、著作権、トレードシークレット(営業秘密)と他の知的所有権に対する権利を保有します。SCILLCの製品/特許の適用対象リストについては、以下のリンクからご覧いただけます。www.onsemi.com/site/pdf/Patent-Marking.pdf。SCILLCは通告なしで、本書記載の製品の変更を行うことがあります。SCILLCは、いかなる特定の目的での製品の適合性について保証しておらず、また、お客様の製品において回路の応用や使用から生じた責任、特に、直接的、間接的、偶発的な損害に対して、いかなる責任も負うことはできません。SCILLCデータシートや仕様書に示される可能性のある「標準的」パラメータは、アプリケーションによっては異なることもあり、実際の性能も時間の経過により変化する可能性があります。「標準的」パラメータを含むすべての動作パラメータは、ご使用になるアプリケーションに応じて、お客様の専門技術者において十分検証されるようお願い致します。SCILLCは、その特許権やその他の権利の下、いかなるライセンスも許諾しません。SCILLC製品は、人体への外科的移植を目的とするシステムへの使用、生命維持を目的としたアプリケーション、また、SCILLC製品の不具合による死傷等の事故が起こり得るようなアプリケーションなどへの使用を意図した設計はされておらず、また、これらを使用対象としておりません。お客様が、このような意図されたものではない、許可されていないアプリケーション用にSCILLC製品を購入または使用した場合、たとえ、SCILLCがその部品の設計または製造に関して過失があったと主張されたとしても、そのような意図せぬ使用、また未許可の使用に関連した死傷等から、直接、又は間接的に生じるすべてのクレーム、費用、損害、経費、および弁護士料などを、お客様の責任において補償をお願いいたします。また、SCILLCとその役員、従業員、子会社、関連会社、代理店に対して、いかなる損害も与えないものとします。SCILLCは雇用機会均等/差別撤廃雇用主です。この資料は適用されるあらゆる著作権法の対象となっており、いかなる方法によっても再販することはできません。