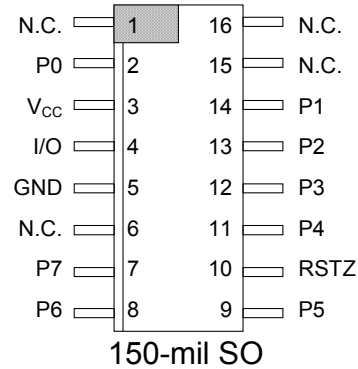


**特長**

- オープンドレイン出力を備える 8 チャンネルのプログラマブル I/O
- PIO プルダウントランジスタのオン抵抗 100Ω(max)、オフ抵抗 10MΩ(typ)
- バスマスタによる問合せ用に、各アクティビティラッチが PIO 入力の非同期状態の遷移を取り込み
- PIO ロジック状態を外付け読み/書き込み回路に同期させるデータストローブ出力
- 内蔵マルチドロップコントローラによって他の Dallas Semiconductor の 1-Wire<sup>®</sup> ネットワーク製品とのコンパチビリティを確保
- プログラマブル PIO 条件で制御する応答で、1-Wire 条件検索コマンドに対応
- 出荷時にレーザ書き込みされたユニークな 64 ビット登録番号で、エラーのないデバイス選択と絶対的な製品 ID を実現
- 1-Wire プロトコルを通じて、15.3kbps または 100kbps で単一デジタル信号によってホストと通信
- 動作範囲: 2.8V ~ 5.25V(-40 ~ +85 )

**端子配置**



**型番**

PART	TEMP RANGE	PACKAGE
DS2408S	-40°C to +85°C	16-Pin SO, 150 mil
DS2408S/ T&R	-40°C to +85°C	Tape-and-Reel of DS2408S

**製品概要**

DS2408 は、8 チャンネル、プログラマブル I/O 1-Wire チップです。PIO 出力はオープンドレイン構造であり、最大 100Ω のオン抵抗です。堅固な PIO チャンネルアクセス通信プロトコルによって、PIO 出力設定の変更がエラーなしで行われます。データ有効ストローブ出力を使って、D/A コンバータ(DAC)やマイクロコントローラのデータバスなどの外付け回路に PIO のロジック状態をラッチすることができます。

DS2408 の動作は、1 線式の 1-Wire バス上で制御します。デバイス通信は、標準 Dallas Semiconductor 1-Wire プロトコルに準拠します。各 DS2408 は、出荷時にチップにレーザ書き込みされた変更不可能でユニークな 64 ビット ROM 登録番号をそれぞれ備えています。この登録番号によってユニークな ID が保証され、この登録番号を使ってマルチドロップ 1-Wire ネットワーク環境でデバイスをアドレス指定することができます。複数の DS2408 デバイスが共通の 1-Wire ネットワーク上で共存し、個別に動作することができます。また、DS2408 は PIO の状態やパワーオンリセット機能に基づいて 1-Wire 条件検索機能にも対応しています。条件検索に加わる条件はプログラマブルです。DS2408 は、オプションの V<sub>CC</sub> 電源接続を備えています。外付け電源がない場合は、デバイスの電源は 1-Wire バスから寄生的に供給されます。外付け電源がある場合は、1-Wire バスの電源がなくても PIO の状態が維持されます。PIO 出力のハードウェア内蔵リセットとして、または PIO の書き込みや読み込みが終了したことを示す外付け回路へのストローブとして機能するように、RSTZ 信号を設定することができます。

1-Wire は Dallas Semiconductor の登録商標です。

**ABSOLUTE MAXIMUM RATINGS\***

P0 to P7, RSTZ, I/O Voltage to GND	-0.5V, +6V
P0 to P7, RSTZ, I/O combined sink current	20mA
Operating Temperature Range	-40°C to +85°C
Junction Temperature	+150°C
Storage Temperature Range	-55°C to +125°C
Lead Temperature (10s)	See J-STD-020A specification

\* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

**ELECTRICAL CHARACTERISTICS**

( $V_{CC} = 0V$  or  $\geq V_{PUP}$ ,  $T_A = -40^\circ C$  or  $+85^\circ C$ .)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
1-Wire Pullup Voltage	$V_{PUP}$	Standard speed	2.8		5.25	V
		Overdrive speed	3.3		5.25	
Standby Supply Current	$I_{CCS}$	$V_{CC}$ at $V_{PUP}$ , I/O pin at 0.3V			1	$\mu A$
<b>I/O Pin General Data</b>						
1-Wire Pullup Resistance	$R_{PUP}$	(Notes 1, 2)			2.2	$k\Omega$
Input Capacitance	$C_{IO}$	(Notes 3, 4)			1200	pF
Input Load Current	$I_L$	I/O pin at $V_{PUP}$ , $V_{CC}$ at 0V			1	$\mu A$
High-to-Low Switching Threshold	$V_{TL}$	(Notes 4, 5, 6)	0.5		3.2	V
Input-Low Voltage	$V_{IL}$	(Notes 1, 7)			0.30	V
Low-to-High Switching Threshold	$V_{TH}$	(Notes 4, 5, 8)	0.8		3.4	V
Switching Hysteresis	$V_{HY}$	(Notes 9, 4)	0.16		0.73	V
Output-Low Voltage at 4mA	$V_{OL}$	(Note 10)			0.4	V
Recovery Time (Note 1)	$t_{REC}$	Standard speed, $R_{PUP} = 2.2k\Omega$	5			$\mu s$
		Overdrive speed, $R_{PUP} = 2.2k\Omega$	2			
		Overdrive speed, Directly prior to reset pulse; $R_{PUP} = 2.2k\Omega$	5			
Rising-Edge Hold-off Time (Notes 11, 4)	$t_{REH}$	Standard speed	0.5		5	$\mu s$
		Overdrive speed	0.5		2	
Timeslot Duration (Notes 1, 12)	$t_{SLOT}$	Standard speed	65			$\mu s$
		Overdrive speed	10			

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
<b>I/O Pin, 1-Wire Reset, Presence-Detect Cycle</b>						
Reset-Low Time (Notes 1, 12)	$t_{RSTL}$	Standard speed, $V_{PUP} > 4.5V$	480		720	$\mu s$
		Standard speed	660		720	
		Overdrive speed	53		80	
Presence-Detect High Time (Note 12)	$t_{PDH}$	Standard speed	15		60	$\mu s$
		Overdrive speed	2		7	
Presence-Detect Fall Time (Note 13)	$t_{FPD}$	Standard speed, $V_{PUP} > 4.5V$	1		5	$\mu s$
		Standard speed	1		8	
		Overdrive speed			1	
Presence-Detect Low Time (Note 12)	$t_{PDL}$	Standard speed, $V_{PUP} > 4.5V$	60		240	$\mu s$
		Standard speed	60		280	
		Overdrive speed	7		27	
Presence-Detect Sample Time (Note 1)	$t_{MSP}$	Standard speed, $V_{PUP} > 4.5V$	65		75	$\mu s$
		Standard speed	68		75	
		Overdrive speed	8		9	
<b>I/O Pin, 1-Wire Write</b>						
Write-0 Low Time (Notes 1, 12)	$t_{W0L}$	Standard speed	60		120	$\mu s$
		Overdrive speed	8		13	
Write-1 Low Time (Notes 1, 12, 14)	$t_{W1L}$	Standard speed	5		$15 - \epsilon$	$\mu s$
		Overdrive speed	1		$1.8 - \epsilon$	
Write Sample Time (Slave Sampling) (Note 12)	$t_{SLS}$	Standard speed	15		60	$\mu s$
		Overdrive speed	1.8		8	
<b>I/O Pin, 1-Wire Read</b>						
Read-Low Time (Notes 1, 15)	$t_{RL}$	Standard speed	5		$15 - \delta$	$\mu s$
		Overdrive speed	1		$1.8 - \delta$	
Read-0 Low Time (Data From Slave) (Note 12)	$t_{SPD}$	Standard speed	15		60	$\mu s$
		Overdrive speed	1.8		8	
Read-Sample Time (Notes 1, 12, 15)	$t_{MSR}$	Standard speed	$t_{RL} + \delta$		15	$\mu s$
		Overdrive speed	$t_{RL} + \delta$		1.8	
<b>P0 to P7, RSTZ Pin</b>						
Input-Low Voltage	$V_{IL}$	(Notes 1, 7)			0.30	V
Input-High Voltage	$V_{IH}$	$V_X = \max(V_{PUP}, V_{CC})$ (Note 1)	$V_X - 0.8$		5.25	V
Output-Low Voltage at 4mA	$V_{OL}$	(Note 10)			0.4	V
Leakage Current	$I_{LP}$	5.25V at the pin			1	$\mu A$
Output Fall Time	$t_{FPIO}$	(Notes 4, 16)	100			ns
Minimum-Sensed PIO Pulse	$t_{PWMIN}$	(Notes 4, 17)	1		5	$\mu s$

- Note 1:** System Requirement
- Note 2:** Maximum allowable pullup resistance is a function of the number of 1-Wire devices in the system and 1-Wire recovery times. The specified value here applies to systems with only one device and with the minimum 1-Wire recovery times. For more heavily loaded systems, an active pullup such as that found in the DS2480B may be required.
- Note 3:** If a 2.2k $\Omega$  resistor is used to pull up the data line to  $V_{PUP}$ , 5 $\mu$ s after power has been applied, the parasite capacitance does not affect normal communications.
- Note 4:** Guaranteed by design—not production tested.
- Note 5:**  $V_{TL}$ ,  $V_{TH}$  are a function of the internal supply voltage.
- Note 6:** Voltage below which, during a falling edge on I/O, a logic '0' is detected.
- Note 7:** The voltage on I/O needs to be less or equal to  $V_{ILMAX}$  whenever the master drives the line low.
- Note 8:** Voltage above which, during a rising edge on I/O, a logic '1' is detected.
- Note 9:** After  $V_{TH}$  is crossed during a rising edge on I/O, the voltage on I/O has to drop by  $V_{HY}$  to be detected as logic '0'.
- Note 10:** The I-V characteristic is linear for voltages less than 1V.
- Note 11:** The earliest recognition of a negative edge is possible at  $t_{REH}$  after  $V_{TH}$  has been reached before.
- Note 12:** Highlighted numbers are NOT in compliance with the published 1-Wire standards. See comparison table below.
- Note 13:** Interval during the negative edge on I/O at the beginning of a presence detect pulse between the time at which the voltage is 90% of  $V_{PUP}$  and the time at which the voltage is 10% of  $V_{PUP}$ .
- Note 14:**  $\epsilon$  represents the time required for the pullup circuitry to pull the voltage on I/O up from  $V_{IL}$  to  $V_{TH}$ .
- Note 15:**  $\delta$  represents the time required for the pullup circuitry to pull the voltage on I/O up from  $V_{IL}$  to the input high threshold of the bus master.
- Note 16:** Interval during the device-generated negative edge on any PIO pin or the RSTZ pin between the time at which the voltage is 90% of  $V_{PUP}$  and the time at which the voltage is 10% of  $V_{PUP}$ . PIO pullup resistor = 2.2k $\Omega$ .
- Note 17:** Width of the narrowest pulse which trips the activity latch (for any PIO pin) or causes a reset (for the RSTZ pin). For a pulse duration  $t_{PW}$ : If  $t_{PW} < t_{PWMIN(min)}$ , the pulse will be rejected. If  $t_{PWMIN(min)} < t_{PW} < t_{PWMIN(max)}$ , the pulse may or may not be rejected. If  $t_{PW} > t_{PWMIN(max)}$  the pulse will be recognized and latched.
- Note 18:** Maximum instantaneous pulldown current through all port pins and the RSTZ pin combined. No requirement for current balance among different pins.

PARAMETER NAME	STANDARD VALUES				DS2408 VALUES			
	STANDARD SPEED		OVERDRIVE SPEED		STANDARD SPEED		OVERDRIVE SPEED	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
$t_{SLOT}$ (incl. $t_{REC}$ )	61 $\mu$ s	(undef.)	7 $\mu$ s	(undef.)	65 $\mu$ s <sup>1)</sup>	(undef.)	10 $\mu$ s	(undef.)
$t_{RSTL}$	480 $\mu$ s	(undef.)	48 $\mu$ s	80 $\mu$ s	660 $\mu$ s	720 $\mu$ s	53 $\mu$ s	80 $\mu$ s
$t_{PDH}$	15 $\mu$ s	60 $\mu$ s	2 $\mu$ s	6 $\mu$ s	15 $\mu$ s	60 $\mu$ s	2 $\mu$ s	7 $\mu$ s
$t_{PDL}$	60 $\mu$ s	240 $\mu$ s	8 $\mu$ s	24 $\mu$ s	60 $\mu$ s	280 $\mu$ s	7 $\mu$ s	27 $\mu$ s
$t_{WOL}$	60 $\mu$ s	120 $\mu$ s	6 $\mu$ s	16 $\mu$ s	60 $\mu$ s	120 $\mu$ s	8 $\mu$ s	13 $\mu$ s
$t_{SLS}$ , $t_{SPD}$	15 $\mu$ s	60 $\mu$ s	2 $\mu$ s	6 $\mu$ s	15 $\mu$ s	60 $\mu$ s	1.8 $\mu$ s	8 $\mu$ s

1) Intentional change, longer recovery-time requirement due to modified 1-Wire front end.

## 端子説明

端子	名称	説明
1	N.C.	未接続
2	P0	チャンネル 0 の I/O 端子。ロジック入力/オープンドレイン出力(オン抵抗は最大 100Ω。動作範囲は 0V ~ 5.25V)。電源オン時のデフォルトは不定です。この出力が「オフ」状態で電源オンするのがアプリケーション上で問題になる場合は、ユーザは適切なパワーオンリセット回路や監視 IC を RSTZ 端子に接続する必要があります。
3	V <sub>CC</sub>	オプションの電源入力。動作範囲は 2.8V ~ 5.25V で、未使用の場合は GND と接続する必要があります。
4	I/O	1-Wire インタフェース。オープンドレインのため、外付けプルアップ抵抗が必要です。
5	GND	グラウンド
6	N.C.	未接続
7	P7	チャンネル 7 の I/O 端子。P0 と同じ特性です。
8	P6	チャンネル 6 の I/O 端子。P0 と同じ特性です。
9	P5	チャンネル 5 の I/O 端子。P0 と同じ特性です。
10	RSTZ	ソフトウェア設定が可能な PIO リセット入力(RST)またはオープンドレインストロープ出力(STRB)。RST として設定されると、ロー入力は、PIO 出力ラッチ状態レジスタの全ビットを設定して、全 PIO 出力を「オフ」状態に設定します。STRB として設定すると、PIO 書込み(「Channel-Access Write」コマンド参照)や PIO 読込み(「Channel-Access Read」)コマンド参照)の後に、出力ストロープが発生します。この端子のパワーオン時のデフォルト機能は RST です。
11	P4	チャンネル 4 の I/O 端子。P0 と同じ特性です。
12	P3	チャンネル 3 の I/O 端子。P0 と同じ特性です。
13	P2	チャンネル 2 の I/O 端子。P0 と同じ特性です。
14	P1	チャンネル 1 の I/O 端子。P0 と同じ特性です。
15	N.C.	未接続
16	N.C.	未接続

## アプリケーション

DS2408 は多用途のデバイスです。標準アプリケーションとして、マイクロコントローラのポートエキスパンダ、リモートマルチチャンネルセンサ/アクチュエータ、マイクロターミナルの通信/制御装置、マイクロコントローラのネットワークインタフェースなどがあります。標準アプリケーション回路及び通信の例は、このデータシートの後半に記載されています(図 17 ~ 22)。

## 概要

図 1 は、DS2408 の主要機能ブロック間の関係を示しています。このデバイスは、以下の 2 つの主要データコンポーネントから構成されています。1) レーザ書込みされた 64 ビット ROM 及び 2) 制御及び状態レジスタの 64 ビットレジスタページ。図 2 は、1-Wire プロトコルの階層構造を示しています。バスマスタは、以下の 8 つの ROM 機能コマンドのいずれか 1 つをまず発行する必要があります。1) Read ROM、2) Match ROM、3) Search ROM、4) Conditional Search ROM、5) Skip ROM、6) Overdrive-Skip ROM、7) Overdrive-Match ROM、または 8) Resume。標準速度で実行される Overdrive ROM コマンドバイトが終了すると、デバイスはオーバドライブモードになり、以後の通信はすべて高速化されます。これらの ROM 機能コマンドに必要なプロトコルは、図 12 に記載されています。ROM 機能コマンドが正常に実行されると、制御機能を利用できるようになり、マスタが 5 つの利用可能コマンドのうちいずれか 1 つを発行することができます。これらの制御コマンドのプロトコルは、図 8 に記載されています。まず最下位ビットから全データの読込みと書込みが行われます。

図 1. DS2408 のブロック図

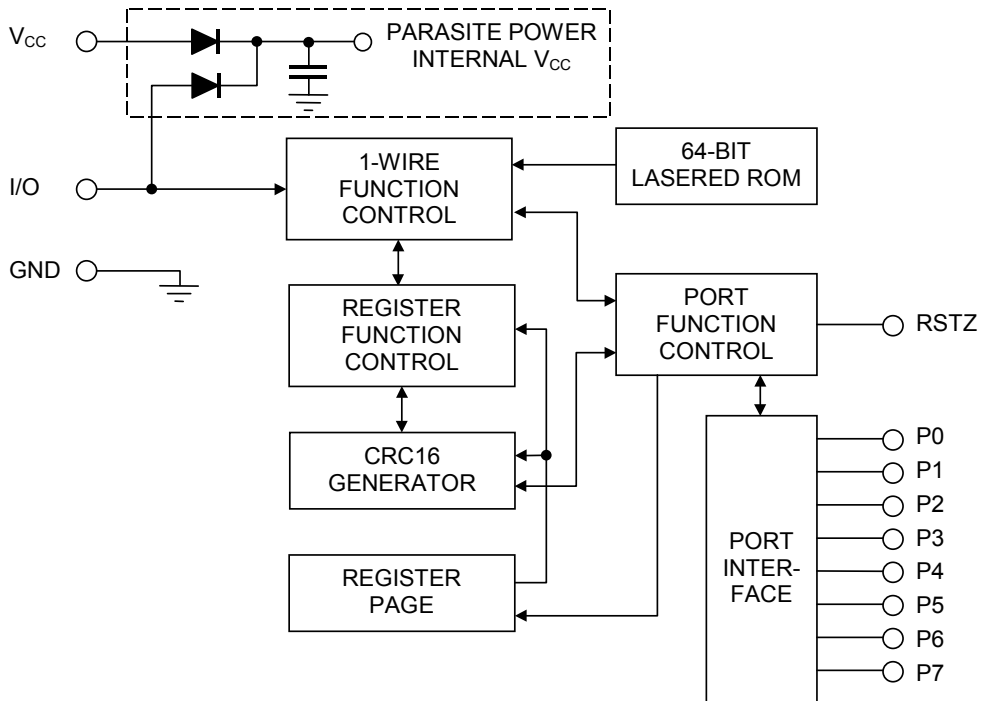
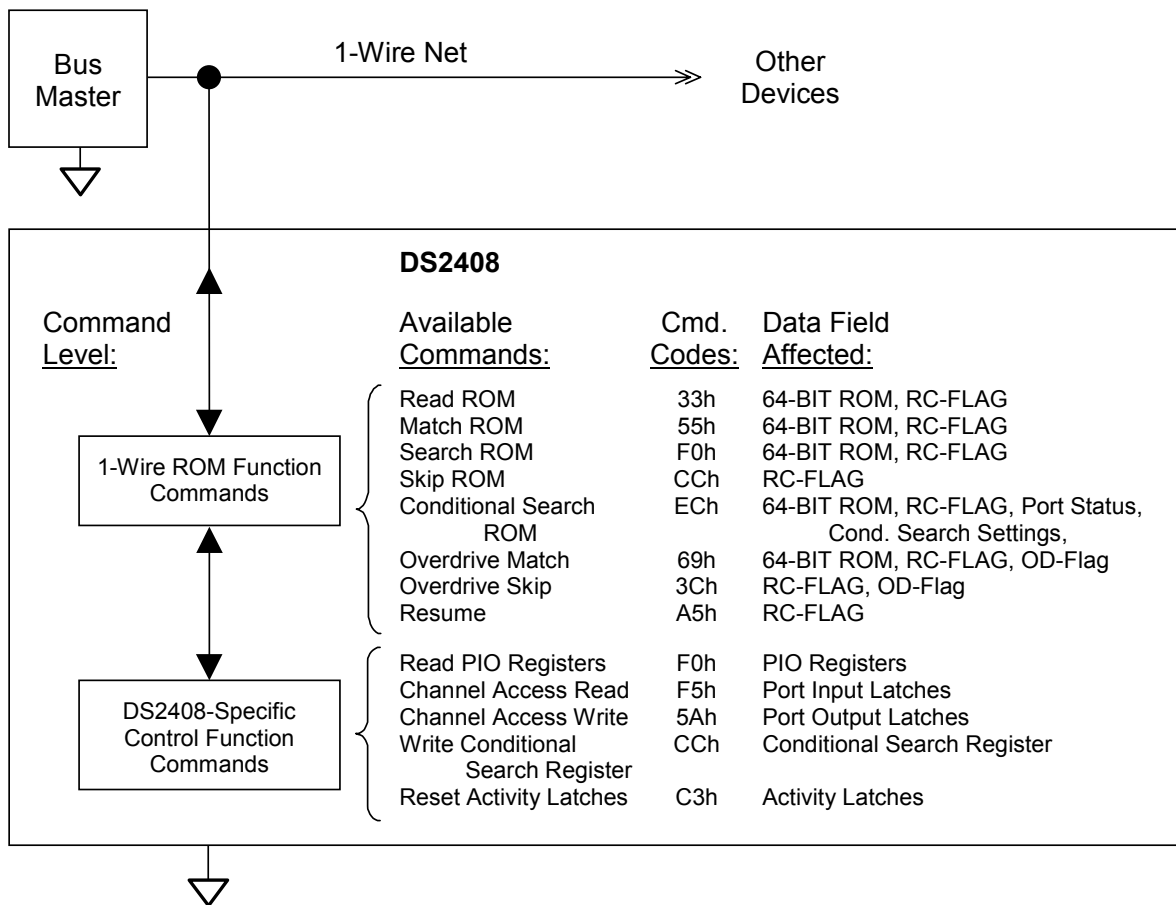


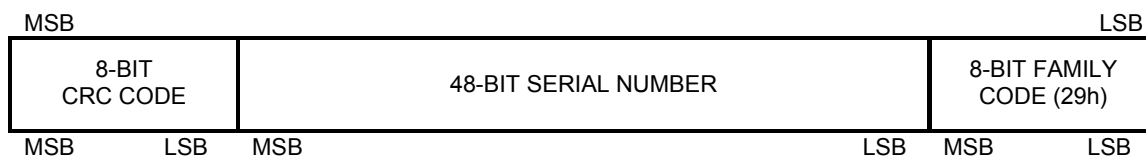
図 2. 1-Wire プロトコルの階層構造



## 寄生電源

DS2408 は信号ラインがハイの間に内蔵コンデンサにエネルギーを保存して、1-Wire バスのみから電源を得ることができます。1-Wire バスがローの間は、1-Wire バスがハイに戻り、寄生(コンデンサ)電源を補充するまで、デバイスはこの「寄生」電源で動作し続けます。電源を利用可能な場合は、V<sub>CC</sub> ピンを外付け電圧電源に接続する必要があります。

図 3. レーザ書き込み 64 ビット ROM

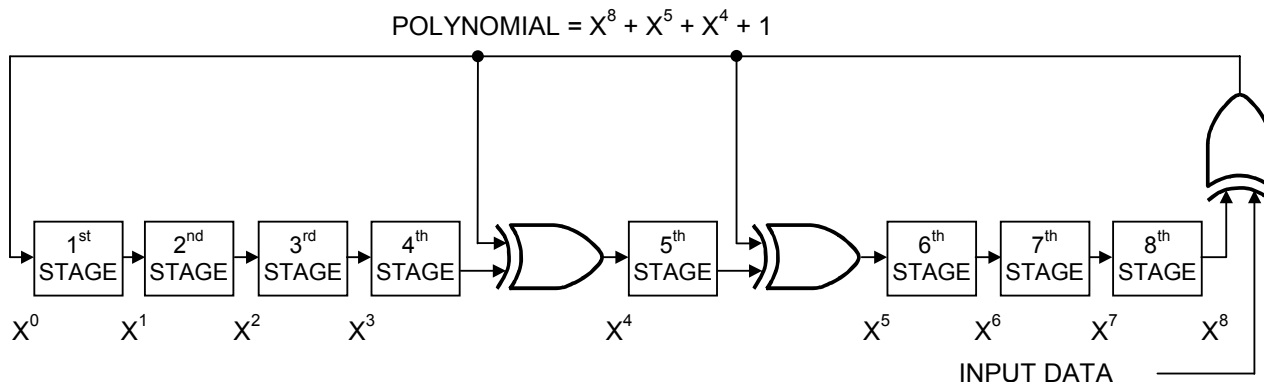


## レーザ書き込み 64 ビット ROM

各 DS2408 は、64 ビット長のユニークな ROM コードを内蔵しています。先頭の 8 ビットは、1-Wire のファミリーコードです。その次の 48 ビットは、ユニークなシリアル番号です。最後の 8 ビットは、先頭から 56 ビットの CRC です。詳細は、図 3 を参照してください。図 4 に示されているように、シフトレジスタと XOR ゲートから構成される生成多項式によって 1-Wire CRC が生成されます。その多項式は、 $X^8 + X^5 + X^4 + 1$  です。Dallas 1-Wire 巡回冗長検査(CRC)に関する詳細は、『アプリケーションノート 27』で利用することができます。

シフトレジスタビットは、0 に初期化されます。次にファミリーコードの最下位ビットから、一度に 1 ビットずつシフトインされます。ファミリーコードの 8 番目のビットが入力されると、シリアル番号が入力されます。シリアル番号が入力されると、シフトレジスタは CRC の値を備えます。したがって、次に CRC コードの 8 ビットをシフトインすると、シフトレジスタはすべて 0 に戻されます。

図 4. 1-Wire CRC 生成式



## レジスタへのアクセス

DS2408 を動作させるのに必要なレジスタは、図 5 に示すようにレジスタページとして構成されています。レジスタはすべて揮発性です。すなわち、デバイスの電源が切断されるとレジスタの状態は消失します。このドキュメントの以降のセクションと図 8 に記載されているデバイス層の *Read PIO Registers* 及び *Write Conditional Search Registers* コマンドを使って、PIO、条件検索、及び制御/状態レジスタの読み込みと書き込みが行われます。

図 5. DS2408 のレジスタアドレスマップ

ADDRESS RANGE	ACCESS TYPE	DESCRIPTION
0000h to 0087h	R	Undefined Data
0088h	R	PIO Logic State
0089h	R	PIO Output Latch State Register
008Ah	R	PIO Activity Latch State Register
008Bh	R/W	Conditional Search Channel Selection Mask
008Ch	R/W	Conditional Search Channel Polarity Selection
008Dh	R/W	Control/Status Register
008Eh to 008Fh	R	These Bytes Always Read FFh

### PIO ロジック状態レジスタ

PIO ピンのロジック状態は、Read PIO Registers コマンドを使ってこのレジスタを読み込んで得られます。このレジスタを読み込むと、 $\overline{\text{STRB}}$ として設定されていても、RSTZ 端子で信号が生成されません。 $\overline{\text{STRB}}$ の詳細については、Channel-Access コマンドの説明を参照してください。

### PIO ロジック状態レジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
0088h	P7	P6	P5	P4	P3	P2	P1	P0

このレジスタは読み専用です。各ビットは、図 6 に示すように各 PIO チャンネル端子に対応しています。このレジスタのデータは、このレジスタの先頭の(最下位)ビットを読み込んでいたバイトの最後の(最上位)ビットでサンプリングされます。詳細は、「Read PIO Registers」コマンドの説明を参照してください。

### PIO 出力ラッチ状態レジスタ

このレジスタのデータは、Channel-access Write コマンドによって PIO に書き込まれた最新データです。このレジスタは、Read PIO Registers コマンドを使って読み込まれます。このレジスタを読み込むと、RSTZ 端子が $\overline{\text{STRB}}$ として設定されていても、その信号が生成されません。 $\overline{\text{STRB}}$ の詳細については、Channel-Access コマンドの説明を参照してください。このレジスタは、デバイスが ESD 発生後に再初期化されても、影響を受けません。

### PIO 出力ラッチ状態レジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
0089h	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0

このレジスタは読み専用です。各ビットは、図 6 に示すように各 PIO チャンネルの出力ラッチに対応しています。

このレジスタのフリップフロップは、電源オン時にランダム状態です。チップが電源オン時に全 PIO チャンネルをオフにして起動する必要がある場合は、ローパルスがオープンドレイン CPU 監視チップなどによって RSTZ 端子に生成される必要があります(図 20 参照)。RC 回路を使ってパワーオンリセットを発生するときは、RSTZ をストロブ出力として設定しないでください(制御/状態レジスタ 008Dh の ROS ビットが 0 である必要があります)。



## PIO アクティビティラッチ状態レジスタ

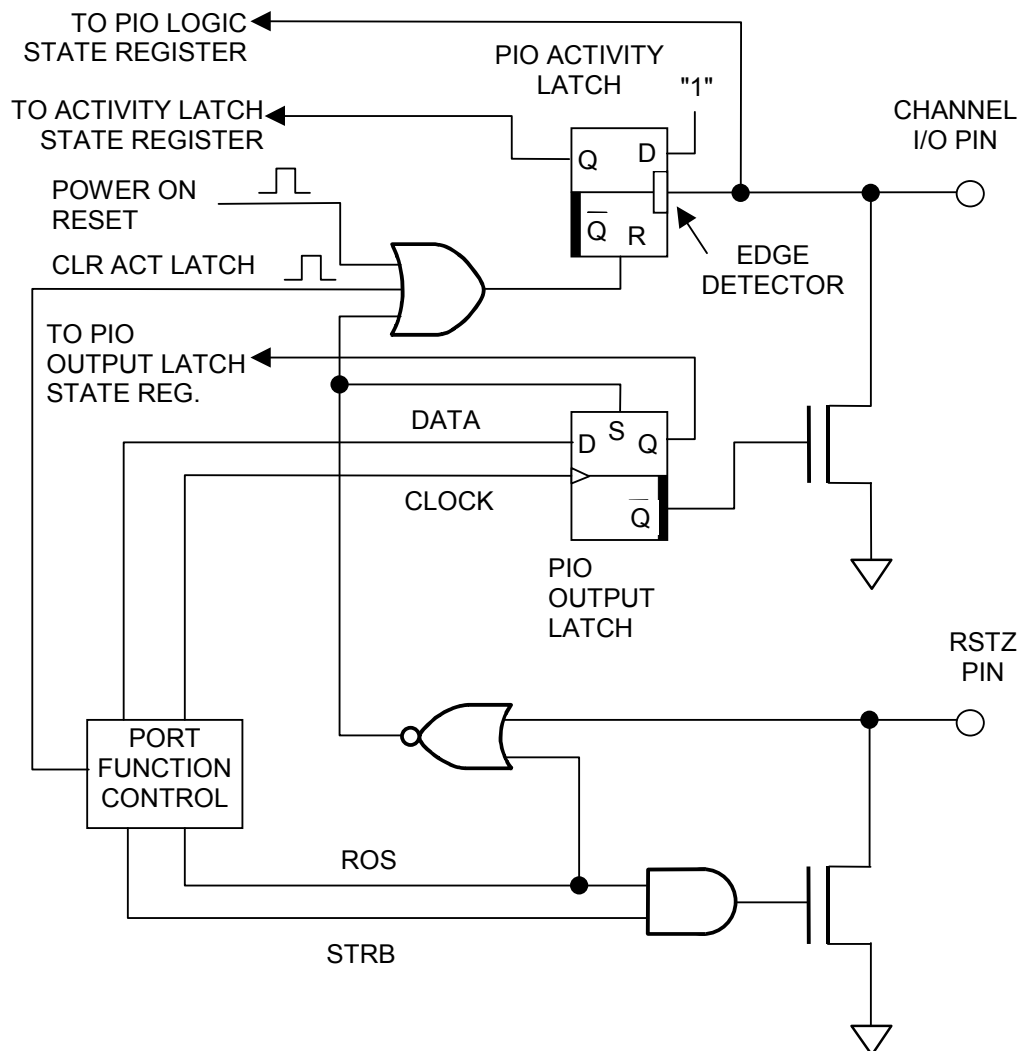
このレジスタのデータは、PIO アクティビティラッチの現状を表しています。このレジスタは、Read PIO Registers コマンドを使って読み込まれます。このレジスタを読み込むと、RSTZ 端子が STRB $\bar$ として設定されていても、その信号が生成されません。STRB $\bar$ の詳細については、Channel-Access コマンドの説明を参照してください。

## PIO アクティビティラッチ状態レジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
008Ah	AL7	AL6	AL5	AL4	AL3	AL2	AL1	AL0

このレジスタは読み専用です。各ビットは、図 6 に示すように各 PIO チャンネルのアクティビティラッチに対応しています。パワーオンリセット、RSTZ 端子のローパルス(RSTZ が RST $\bar$  入力として設定されている場合のみ)、または Reset Activity Latches コマンドの正常実行によって、このレジスタは 00h にクリアされます。

図 6. チャンネル I/O 及び RSTZ 簡略ロジック図



## 条件検索チャンネル選択マスクレジスタ

このレジスタのデータは、PIO チャンネルが条件検索コマンドに加わるのに適合しているかどうかを制御します。1 つまたは複数の PIO チャンネルを組み込むには、こうしたチャンネルに対応するこのレジスタのビットを 1 に設定する必要があります。Write Conditional Search Registers コマンドでのみ、このレジスタに書き込むことができます。

## 条件検索チャンネル選択マスクレジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
008Bh	SM7	SM6	SM5	SM4	SM3	SM2	SM1	SM0

このレジスタは読み込み及び書き込み用です。各ビットは、図 7 に示すように各 PIO チャンネルに対応しています。このレジスタは、パワーオンリセットで 00h にクリアされます。

## 条件検索チャンネル極性選択レジスタ

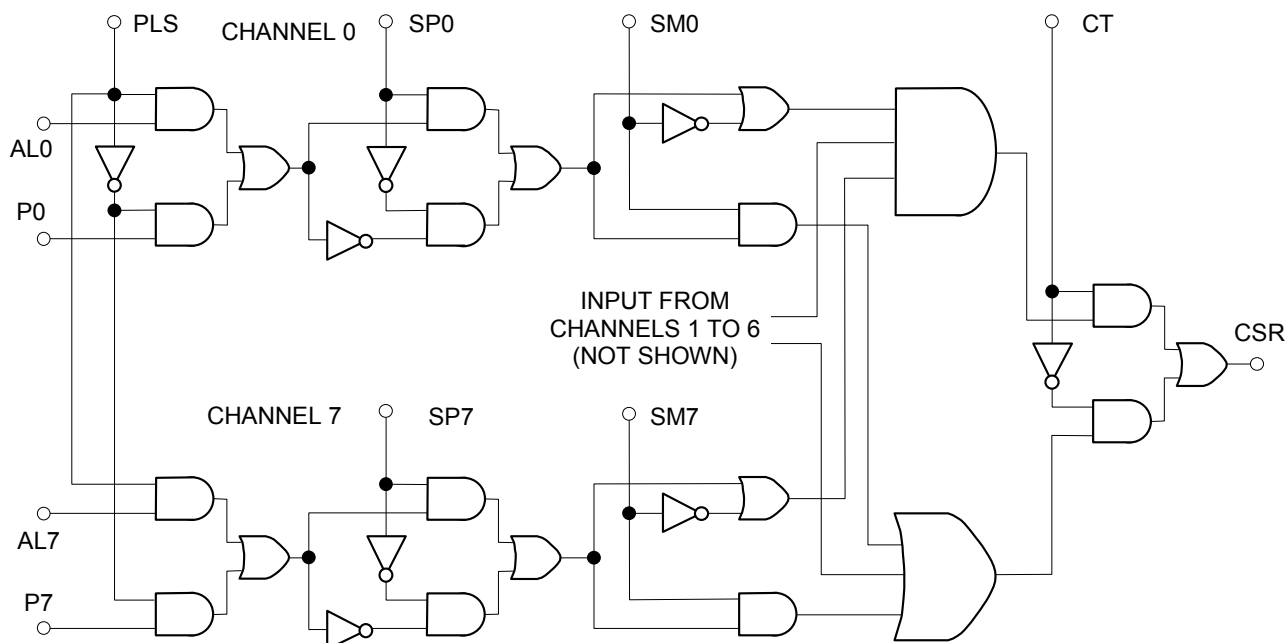
このレジスタのデータによって、デバイスが条件検索コマンドにตอบสนองするように、選択した各 PIO チャンネルの極性を設定します。PIO チャンネル内で、アドレス 008Dh の制御/状態レジスタにある PLS ビットの指定によって、データソースをチャンネルの入力信号(端子)またはチャンネルのアクティビティラッチのいずれかにすることができます。Write Conditional Search Registers コマンドでのみ、このレジスタに書き込むことができます。

## 条件検索チャンネル極性選択レジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
008Ch	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0

このレジスタは読み込み及び書き込み用です。各ビットは、図 7 に示すように各 PIO チャンネルに対応しています。このレジスタは、パワーオンリセットで 00h にクリアされます。

## 図 7. 条件検索のロジック



## 制御/状態レジスタ

このレジスタのデータは状態の情報を伝え、RSTZ ピンの機能を設定して、さらにデバイスを条件検索に設定します。Write Conditional Search Registers コマンドでのみ、このレジスタに書き込むことができます。

## 制御/状態レジスタビットマップ

ADDR	b7	b6	b5	b4	b3	b2	b1	b0
008Dh	VCCP	0	0	0	PORL	ROS	CT	PLS

このレジスタは読み込み及び書き込み用です。V<sub>CC</sub> 電源がない場合は、このレジスタはパワーオンリセット後に 08h を読み込みます。各ビットの機能割当ては、下表で説明されています。ビット 4～6 は機能を備えていません。これらのビットは常に 0 を読み込み、1 に設定することはできません。

## 制御/状態レジスタの詳細

ビット説明	ビット	設定
PLS: 端子またはアクティビティラッチの選択	ビット 0	PIO 端子または PIO アクティビティラッチを条件検索の入力として選択します。 0: PIO 端子を選択(デフォルト) 1: アクティビティラッチを選択
CT: 条件検索ロジック項	ビット 1	デバイスが条件検索に回答する適合条件を満たすために、複数チャンネルのデータの OR や AND をとる必要があるかどうかを設定します。単一チャンネルのみがチャンネル選択マスク(008Bh)で選択されている場合は、このビットは設定不要です。 0: ビットワイズ OR(デフォルト) 1: ビットワイズ AND
ROS: RSTZ 端子モード制御	ビット 2	RSTZ を入力または $\overline{\text{STRB}}$ 出力として設定します。 0: $\overline{\text{RST}}$ 入力として設定(デフォルト) 1: $\overline{\text{STRB}}$ 出力として設定
PORL: パワーオンリセットラッチ	ビット 3	デバイスがパワーオンリセットを実行するかを設定します。ソフトウェア制御下でのみこのビットを 0 にクリア可能です。このビットが 1 である限り、デバイスは常に条件検索に回答します。
VCCP: V <sub>CC</sub> 電源状態 (読み込み専用)	ビット 7	V <sub>CC</sub> 駆動の動作の場合は、V <sub>CC</sub> 端子を電圧源 $\geq V_{\text{PUP}}$ に接続する必要があります。 0: V <sub>CC</sub> 端子をグランド 1: V <sub>CC</sub> 駆動動作

デバイスが条件検索に回答するかどうかを設定する各信号の相互関係が図 7 に示されています。選択マスクの SM によって参加するチャンネルを選択します。極性選択の SP で、条件検索に適合させるために、必要に応じてチャンネル信号を 1 または 0 にチャンネルごとに設定します。PLS ビットによって、全チャンネル信号がアクティビティラッチあるいは I/O ピンから取得されるかを設定します。全チャンネルの信号は、AND ゲート及び OR ゲートに入力されます。CT ビットによって、最後に AND(論理積)または OR(論理和)された結果を条件検索応答信号 CSR として選択します。

## CT ビットに関する注記

OR 選択したチャンネルの 1 つ以上の入力(端子状態またはアクティビティラッチ)が対応する極性と一致すると、適合条件が満たされます。

AND 適合条件を満たすには、選択したすべてのチャンネルの入力(ピン状態またはアクティビティラッチ)が対応する極性と一致する必要があります。

図 8-1. 制御機能フローチャート

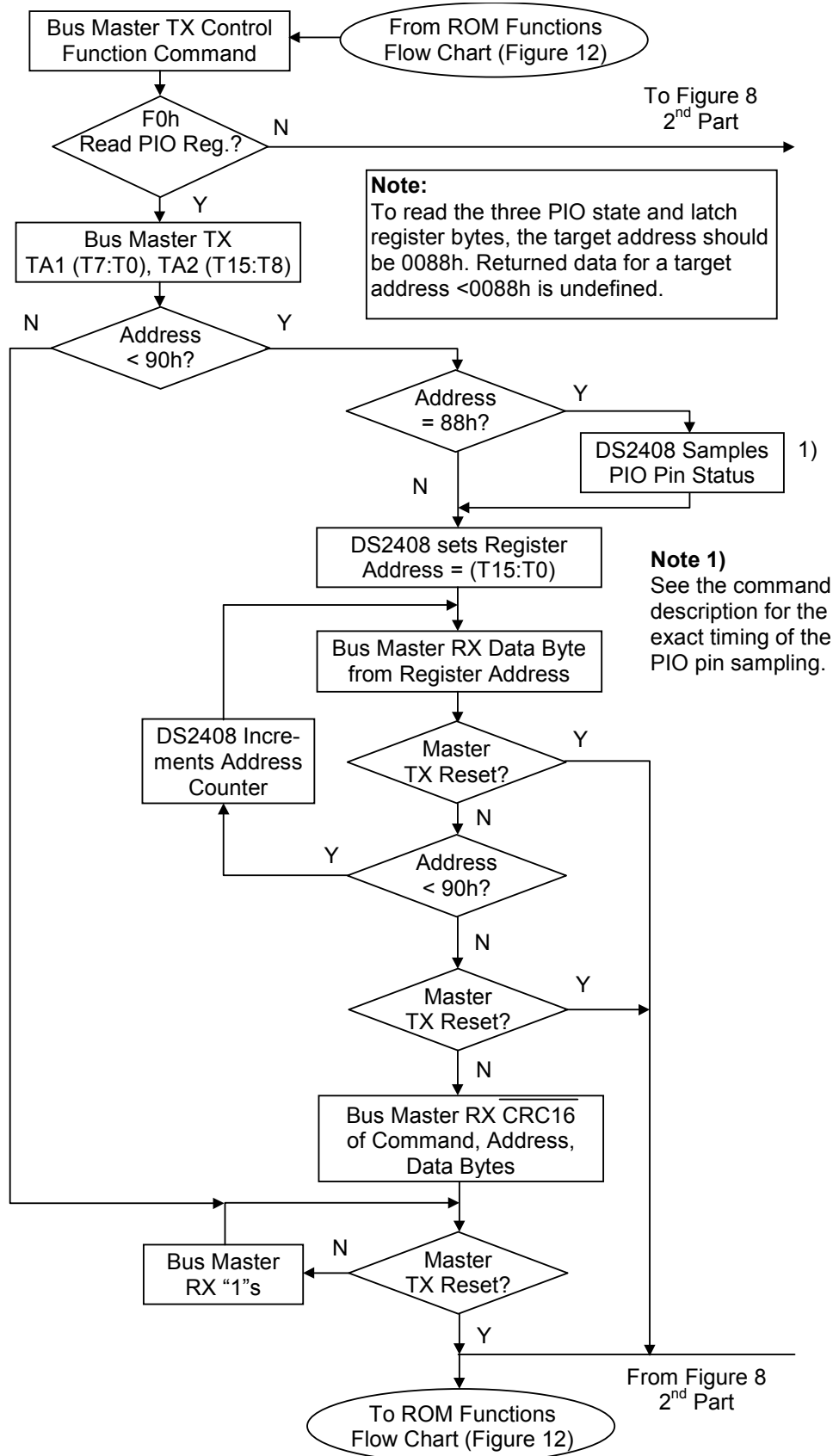


図 8-2. 制御機能フローチャート

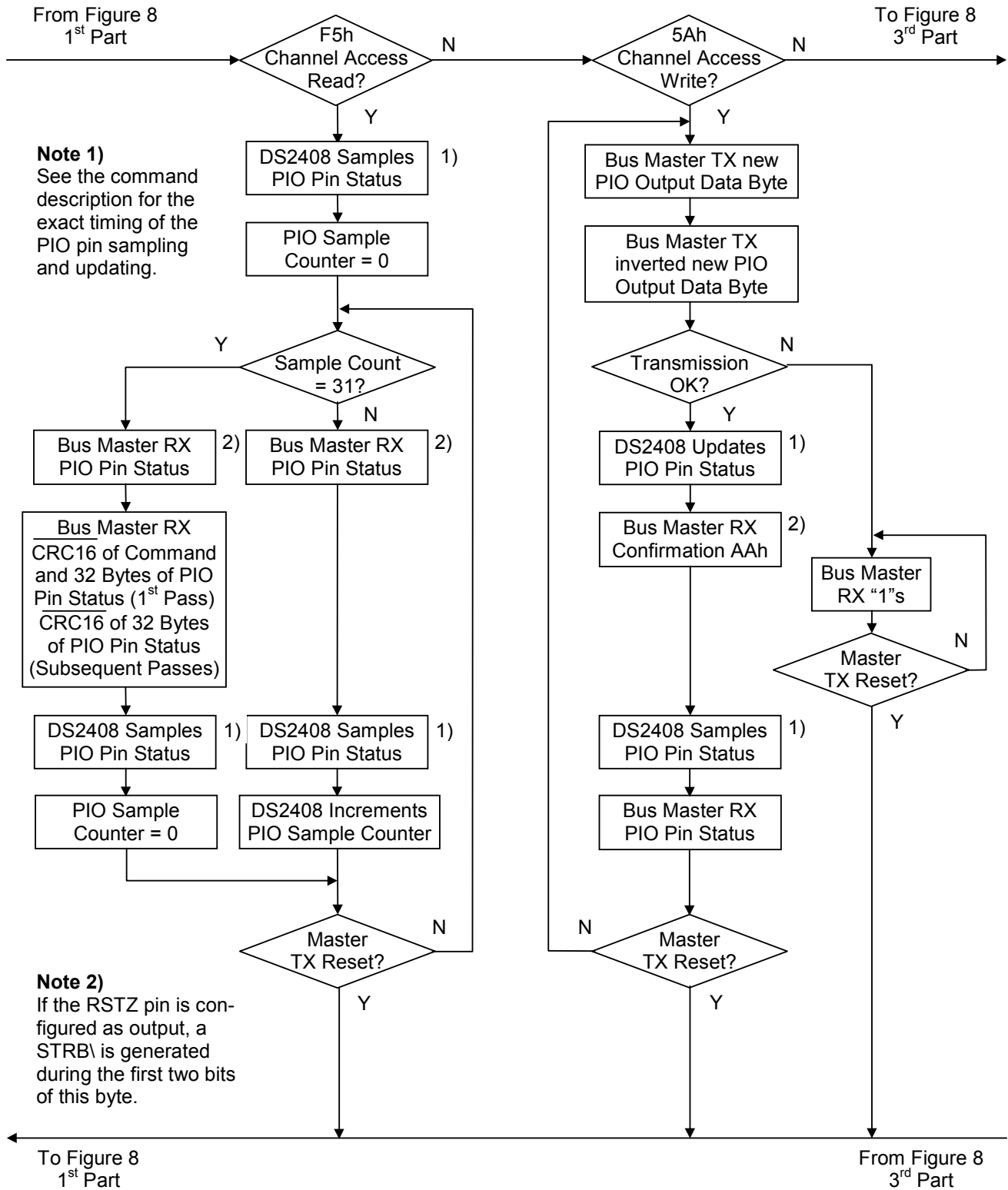
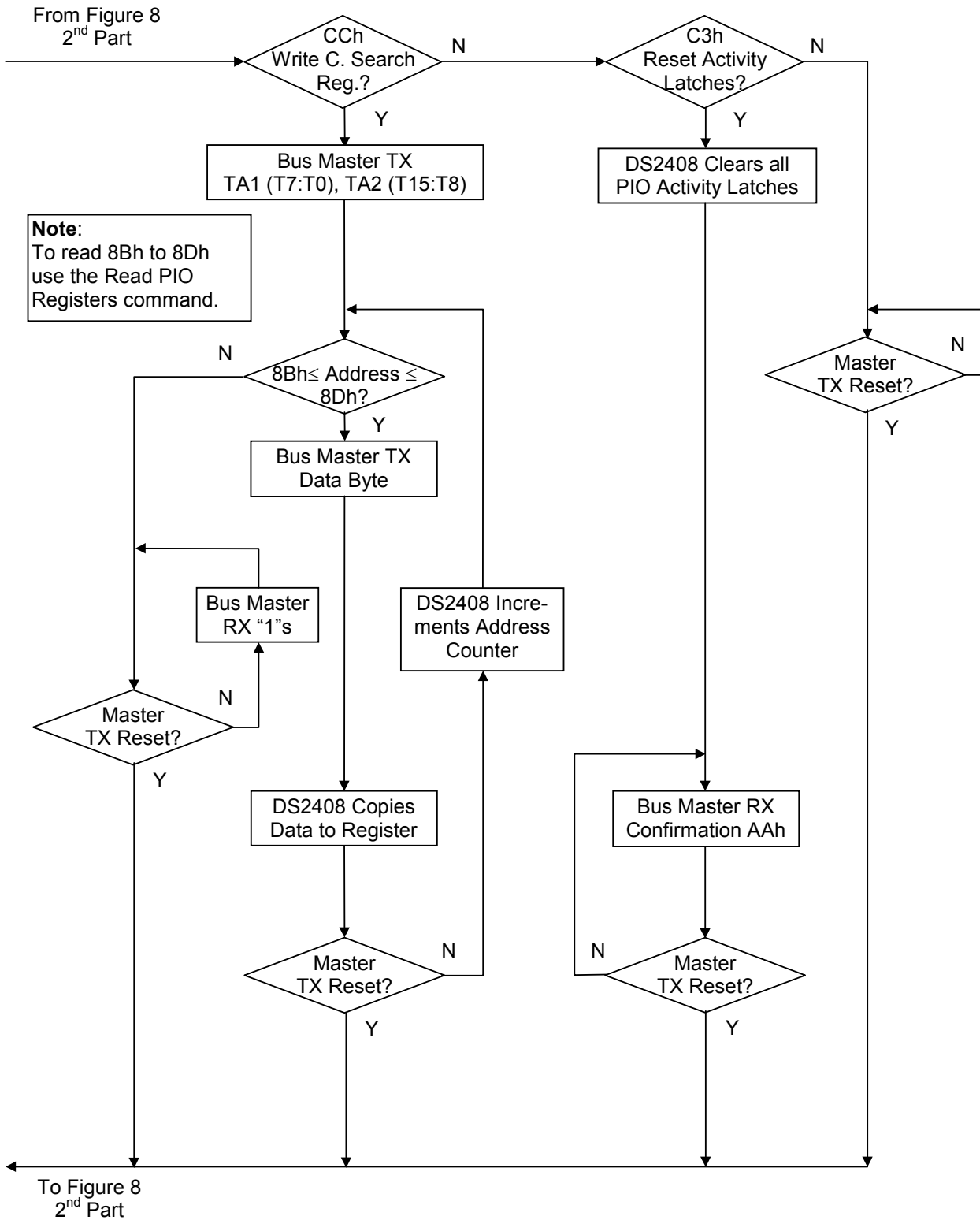


図 8-3. 制御機能フローチャート



## 制御機能コマンド

ROM 機能コマンドが完了すると、制御機能コマンドを発行することができます。「制御機能フローチャート」(図 8)は、PIO チャンネル及び DS2408 の特別機能レジスタへのアクセスに必要なプロトコルを示しています。マスタと DS2408 間の通信は、標準速度(デフォルト、OD = 0)、またはオーバドライブ速度(OD = 1)のいずれかで行われます。オーバドライブモードに明確に設定されていない場合は、デバイスは標準速度で動作します。

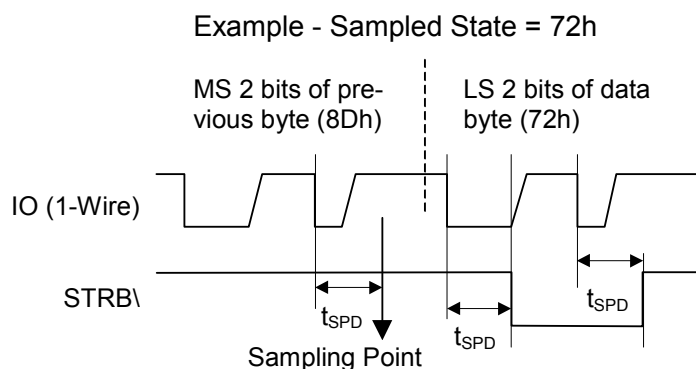
### Read PIO Registers [F0h]

Read PIO Registers コマンドを使って、デバイスのどのレジスタも読み込むことができます。コマンド発行後に、マスタは 2 バイトのターゲットアドレスを提供する必要があります。これらの 2 バイトの後に、マスタはターゲットアドレスから始まるデータを読み込み、アドレス 008Fh まで続けることができます。マスタが読み込みを継続する場合は、コマンド、アドレスバイト、及びレジスタページの先頭の開始バイトから最後まで読み込まれた全データバイトの反転 16 ビット CRC を受け取ります。この CRC16 は CRC 生成式をクリアし、コマンドバイトに続いて、2 バイトのアドレスバイト、レジスタページのアドレス指定された先頭位置から始まり、最後のバイトまで続くデータバイトをシフトインした結果です。バスマスタが CRC16 を受け取ると、1-Wire Reset コマンドが発行されるまで、DS2408 はロジック 1 を以降の読み込みタイムスロットで返します。このコマンドがターゲットアドレス 0088h(PIO ロジック状態レジスタ)で発行される場合は、PIO のサンプリングが TA2 の最上位ビットの送信時に行われます。ターゲットアドレスが 0088h より下位の場合は、マスタがアドレス 0087h から最上位ビットを読み込む間、PIO サンプリングが行われます。

### Channel-Access Read [F5h]

アドレス 88h から PIO ロジック状態を読み込むのとは対照的に、このコマンドは永久ループで PIO ロジック状態を読み込みます。32 バイトの PIO ピン状態の後に、DS2408 は反転 CRC16 をデータストリームに挿入します。これによって、マスタはデータがエラーなしで受信されたかを検証することができます。Channel-Access Read は、いつでも 1-Wire Reset で終了することができます。

## 図 9. Channel-Access Read タイミング



### 注記:

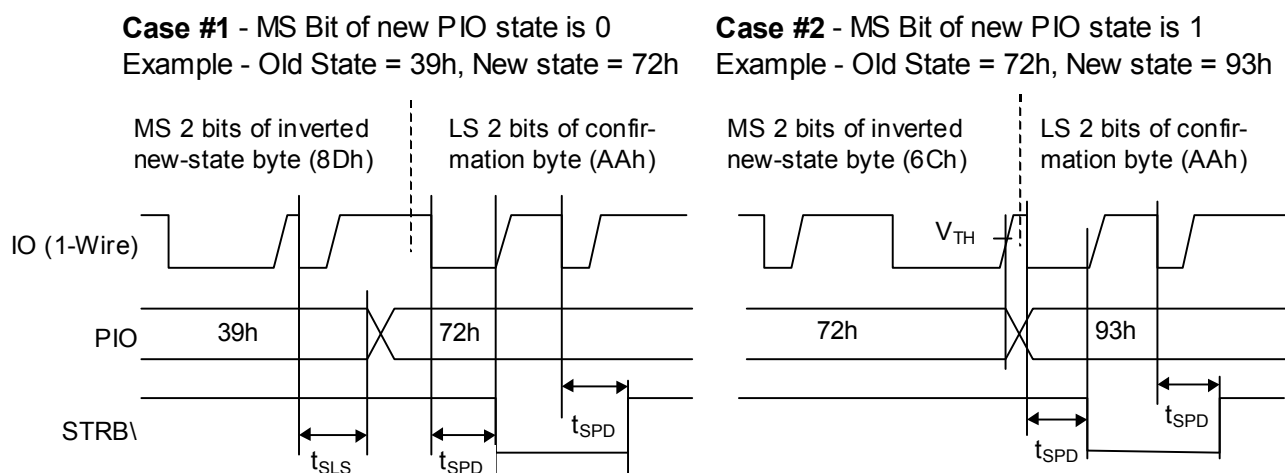
- 1) 「直前のバイト」はコマンドコード、直前の PIO サンプリングによってもたらされたデータバイト、または CRC16 の上位バイトになる場合もあります。この例は、read-1 タイムスロットを示しています。
- 2) 「直前のバイト」が書き込み確認バイト(AAh)の場合、サンプリングポイントのタイミングが Channel-access Write コマンドにも適用されます。Channel-Access Write コマンドの間にサンプリングが行われると、 $\overline{\text{STRB}}$  パルスは発生しません。

8 つの全 PIO チャンネルの状態が同時にサンプリングされます。コマンドコード F5h の最後の(最上位)ビットの間に、最初のサンプリングが行われます。マスタが PIO の状態の MSB(すなわち、ピン P7 の状態)を受け取る間に、次のサンプリングが行われ、マスタが 31 の PIO サンプルを受け取るまで、以下同様に実行されます。次に、マスタはコマンドバイト及び 32 の PIO サンプル(最初のパス)の反転 CRC16、または 32 の PIO サンプル(以降のパス)の反転 CRC を受け取ります。CRC の最後(最上位)のビットが送信される間に、次の PIO のサンプリングが行われます。タイムスロットの開始とサンプリングポイントと間の遅延は、送信されるビット値とデータ方向とは無関係です(図 9 参照)。RSTZ 端子を STRB $\bar{}$ として設定すると、PIO データの先頭の 2(最下位)ビットの送信時にストロブ信号が生成されます。このストロブ信号は、FIFO やマクロコントローラに対して、マスタが 1-Wire ラインを通じて次の PIO データバイトを読み込むように指示できます。

### Channel-Access Write [5Ah]

Channel-Access Write コマンドは PIO 出力ラッチ状態レジスタ(address 0089h)に書き込む唯一の方法です。このレジスタは PIO チャンネルのオープンドレイン出力トランジスタを制御します。永久ループで、このコマンドはまず PIO に新規データを書き込んでから、PIO の状態をリードバックします。マスタが暗黙のリードアフタライトを用いて、状態を検証したり、ポート端子及び同期用の RSTZ と接続したマイクロコントローラと高速通信することができます。Channel-Access Write は、いつでも 1-Wire Reset で終了することができます。

### 図 10. Channel-Access Write タイミング



### 注記:

両例は、RSTZ 端子が STRB $\bar{}$  出力として設定されていることを前提としています。RSTZ が RST 入力(デフォルト)として設定されている場合は、Channel-Access Write が正常に機能するように、RSTZ 端子をハイに( $V_{CC}$ または  $V_{PUP}$ に)接続する必要があります。このピンをフローティング状態にすると、PIO チャンネルの出力トランジスタが「オフ」状態になり、すべての PIO 出力ラッチが「1」を読み込みます。このロジック図については、図 6 を参照してください。

コマンドコードの後に、マスタは PIO 出力トランジスタの新たな状態を設定するバイトを送信します。先頭の(最下位)ビットは、P0 に対応しています。出力トランジスタをオフ(非導通にするには、該当ビット値を 1 にします。トランジスタをオンにするには、このビットは 0 である必要があります。これにより、新たな PIO 出力状態として送信されたデータバイトは、true(非反転)形式で PIO 端子に到達します。送信をデータエラーから保護するには、マスタは反転形式で新たな PIO バイトを繰り返す必要があります。送信が成功した場合のみ、PIO の状態が変化します。図 10 に示すように、PIO の新たな状態への実際の遷移は新規反転 PIO データバイトの最後の(最上位)ビットの間に行われ、このビットの極性に依存します。このビットが 1 の場合は、 $t_{SLS}$  の終了後に遷移が始まります。0 の場合は、タイムスロットの最後でスレッショルド  $V_{TH}$  を超えると遷移が始まります。PIO の状態の正常な変化をマスタに通知するために、DS2408 はデータパターンが



AAh の確認バイトを送信します。RSTZ ピンを  $\overline{\text{STRB}}$  として設定されている場合は、確認バイトの先頭の 2(最下位)ビットの送信時にストロブ信号が生成されます。このストロブ信号は、FIFO やマイクロコントローラに対して、PIO から新規データバイトを読み込むように指示します。確認バイトの最後のビットが送信される間に、図 9 に示すように DS2408 は PIO 端子の状態をサンプリングし、その状態をマスタに送信します。このデータに応じて、マスタは PIO にさらにデータを書込み続けるか、または 1-Wire リセットを発行してコマンドを終了することができます。

### Write Conditional Search Register [CCh]

このコマンドは、デバイスが Conditional Search コマンドに応答するために満たす必要がある条件を DS2408 に伝える、RSTZ 端子の機能を設定する、及びパワーオンリセットフラグをクリアするのに使用します。

コマンド発行後に、マスタは 2 バイトのターゲットアドレスを送信します。このアドレスは、008Bh ~ 008Dh の値である必要があります。次に、マスタは書き込むバイトをアドレス指定されたセルに送信します。アドレスが有効な場合は、このバイトはレジスタページのそのアドレス位置にすぐに書き込まれます。ここで、マスタは 1-Wire リセットを発行してコマンドを終了するか、あるいは次の上位アドレスに別のバイトを送信することができます。レジスタアドレス 008Dh に書き込まれると、以降のデータバイトは無視されます。マスタは 1-Wire リセットを送信して、コマンドを終了する必要があります。Write Conditional Search Register のフローには新しいレジスタデータのエラーチェックがないので、Read PIO Registers コマンドを使ってレジスタを読み込んで書き込みが正しいかどうかを検証する必要があります。

### Reset Activity Latches [C3h]

各 PIO チャンネルは、PIO ピンで状態変化があるごとに設定されるアクティビティラッチを備えています。この変化は、外部のイベント/信号や、PIO への書き込みによって発生します。アプリケーションに応じて、外部イベントを取り込み、処理した後に、アクティビティラッチのリセットを要する場合があります。PIO Activity Latch State Register には読み込みのアクセスのみあるので、DS2408 にはラッチをリセットする特別なコマンドがあります。コマンドコードを受け取った後に、デバイスは全アクティビティラッチを同時にリセットします。マスタが Reset Activity Latches コマンドの実行を検証するには 2 つの方法があります。最も簡単な方法は、コマンドコードが送信された直後に 1-Wire ラインから読み込みを開始する方法です。この場合は、マスタが 1-Wire リセットを送信するまで、AAh バイトを読み込みます。実行を検証するもう 1 つの方法は、レジスタアドレス 008Ah を読み込むことです。

## 1-Wire バスシステム

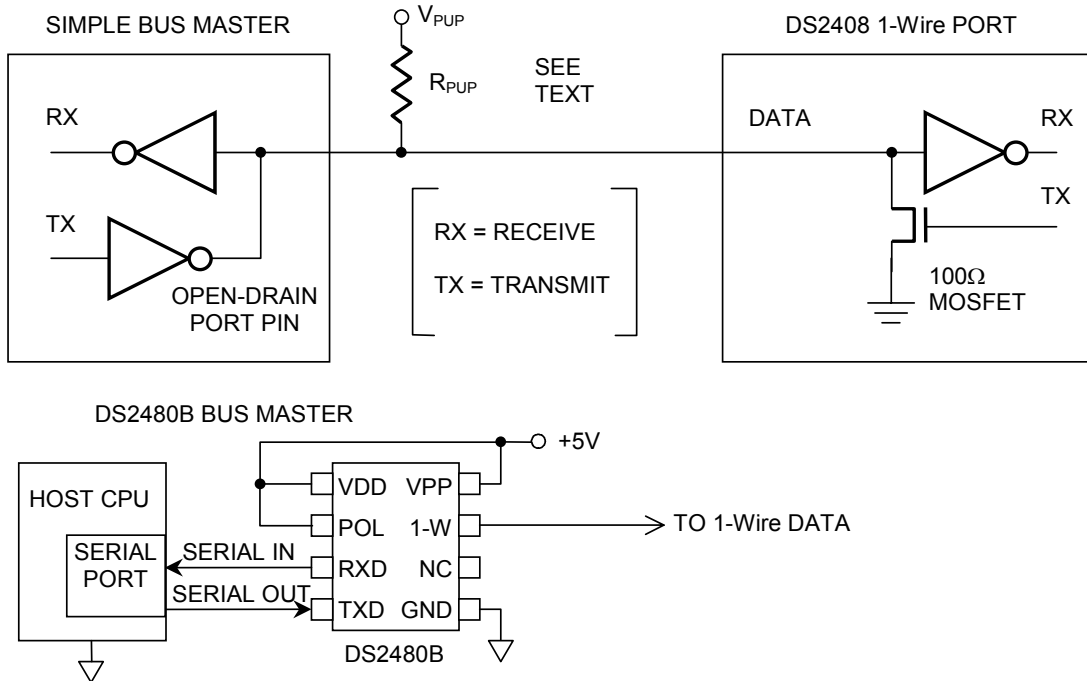
1-Wire バスは、単一のバスマスタと、1 つまたは複数のスレーブから構成されるシステムです。どの場合においても、DS2408 はスレーブデバイスです。バスマスタは、通常マイクロコントローラか PC です。小規模構成の場合は、単一ポート端子を使って 1-Wire 通信信号をソフトウェア制御の下で生成することができます。マルチセンサネットワークの場合は、DS2480B 1-Wire ラインドライバデバイスや、当デバイスをベースにしたシリアルポートアダプタ(DS9097U シリーズ)を推奨します。これによってハードウェア設計が容易になり、マイクロプロセッサがリアルタイムの応答から解放されます。

このバスシステムの説明は、以下の 3 つのトピックに分類されます。すなわち、ハードウェア構成、トランザクションシーケンス、及び 1-Wire 信号方式(信号種類とタイミング)です。1-Wire プロトコルでは、バスマスタの同期パルスの立下りで起動した特定タイムスロット時のバス状態によってバストランザクションを定義します。

## ハードウェア設定

1-Wire バスはその名の通り 1 つのラインしか持っていません。バス上の各デバイスが適時このラインを駆動することができる必要があります。これを容易にするために、1-Wire バスに接続された各デバイスがオープンドレイン出力やトライステート出力を備える必要があります。DS2408 の 1-Wire ポートは、図 11 に示されているのと等価の内蔵回路付オープンドレインです。

図 11. ハードウェア構成



マルチドロップバスは、複数のスレーブを接続した 1-Wire バスから構成されています。標準速度では 1-Wire バスの最高データ速度は、15.3kbps です。1-Wire デバイスの通信速度は、オーバドライブモードを作動させて 142kbps まで一般に向上させることができます。ただし、DS2408 の最高オーバドライブデータ速度は 100kbps です。プルアップ抵抗の値は、主にネットワーク規模と負荷条件に左右されます。ほとんどのアプリケーションの場合は、プルアップ抵抗の最適値は標準速度で約 2.2kΩ で、オーバドライブ速度で 1.5kΩ です。

1-Wire バスのアイドル状態はハイです。なんらかの理由でトランザクションを一時停止する必要があり、その後トランザクションが再開する場合は、バスがアイドル状態になっている必要があります。これが実行されず、バスが 16μs (オーバドライブ速度)、または 120μs (標準速度) を超えてローにされると、バス上の 1 つまたは複数のデバイスがリセットされることがあります。DS2408 の場合は、1-Wire バス上のスレーブデバイスがリセットを実行しないように、バスはオーバドライブ速度で 13μs を超えてロー状態にしてはいけません。DS2480B 1-Wire ドライバ及び当ドライバチップをベースとしたシリアルポートアダプタと併用されると、DS2408 は正常に通信します。オーバドライブや 4.5V 以下でデバイスを動作させるときは、一部の 1-Wire I/O タイミング値を変更する必要があります(EC 表参照)。

## トランザクションシーケンス

1-Wire ポートを通じて DS2408 にアクセスするプロトコルは、以下の通りです。

- 初期化
- 機能コマンド
- 制御機能コマンド
- トランザクション/データ

各制御機能コマンドのトランザクションシーケンスの説明は、このドキュメントの後半にあります。

### 初期化

1-Wire バス上のすべてのトランザクションは、初期化シーケンスから始まります。初期化シーケンスはバスマスタから送信されるリセットパルスで構成され、このリセットパルスの後には(単数または複数の)スレーブが送信する(単数または複数の)プレゼンスパルスが続きます。プレゼンスパルスによって、バスマスタは DS2408 がバス上にあり、動作可能であることを認識します。詳細は、「1-Wire 信号方式」の項を参照してください。

### ROM 機能コマンド

バスマスタがプレゼンスパルスを検出すると、バスマスタは 7 つの ROM 機能コマンドのいずれか 1 つを発行することができます。ROM 機能コマンドはすべて、8 ビット長です。こうしたコマンドのリストは、以下の通りです(図 12 のフローチャート参照)。

#### Read ROM [33h]

このコマンドを使うと、バスマスタが DS2408 の 8 ビットファミリコード、ユニークな 48 ビットシリアル番号、及び 8 ビット CRC を読み込むことができます。バス上に単一デバイスがある場合のみ、このコマンドを使用することができます。複数のスレーブがバス上にある場合は、全スレーブが同時に送信しようとする、データの衝突が発生します(オープンドレインにより、ワイヤード AND 結果がもたらされます)。結果としてもたらされるファミリコードと 48 ビットシリアル番号によって、CRC の不整合が発生します。

#### Match ROM [55h]

64 ビット ROM シーケンスが後に続く Match ROM コマンドを使って、バスマスタがマルチドロップバス上の特定 DS2408 をアドレス指定することができます。64 ビット ROM シーケンスと完全一致する DS2408 のみが、以下の制御機能コマンドに応答します。64 ビット ROM シーケンスと一致しないスレーブはすべて、リセットパルスを待ちます。バス上の 1 つまたは複数のデバイスで、このコマンドを使用することができます。

#### Search ROM [F0h]

システムが最初に立ち上がる際に、バスマスタが 1-Wire バス上のデバイス数やデバイスの 64 ビット ROM コードを認識しない場合があります。Search ROM コマンドによって、バスマスタは消去プロセスを使って、バス上の全スレーブデバイスの 64 ビット ROM コードを識別することができます。Search ROM プロセスは、以下のシンプルな 3 ステップのルーチンの反復です。1 ビットを読み込み、当ビットの補数を読み込んで、当ビットの任意の値を書き込む。バスマスタは、ROM の各ビットでこのシンプルな 3 ステップのルーチンを実行します。こうしたプロセスの後に、バスマスタは 1 デバイス内の ROM の内容を認識します。残りのデバイス数とデバイスの ROM コードは、プロセスを重ねることによって識別することができます。ソフトウェア例などの Search ROM コマンドプロセスの詳細については、『アプリケーションノート 187』を参照してください。

#### Conditional Search [ECh]

Conditional Search ROM コマンドは、特定条件を満たすデバイスのみが検索に加わることを除いて、Search ROM コマンドと同様に機能します。条件は、条件検索チャンネル及び極性選択(アドレス 008Bh、008Ch)、制御/状態レジスタ(アドレス 008Dh)のビット機能 CT 及び PLS、ならびに PIO チャンネルの状態によって指定されます。条件検索ロジックに

については、図 7 を参照してください。また、デバイスは、PORL ビットが設定されている場合は、条件検索に応答します。Conditional Search ROM を使って、外部信号による PIO ピンの状態変化など重要イベントを通知する必要があるマルチドロップシステム上のデバイスをバスマスタが効率的に識別することができます。条件検索の各プロセスがマルチドロップバス上の特定デバイスの 64 ビット ROM を適切に識別すると、まるで Match ROM が発行されたかのように当該デバイスに個別にアクセスすることができます。というのは、他の全デバイスは検索プロセスから抜け、リセットパルスに対して待機しているからです。

### Skip ROM [CCh]

バスマスタが 64 ビット ROM コードを提供せずに制御機能にアクセスできるようにして、このコマンドを使って単ドロップバスシステムで時間を削減することができます。複数のスレーブがバス上にあり、Skip ROM コマンドに続き Read コマンドが発行される場合に、複数のスレーブが同時送信すると、データの衝突がバス上に発生します(オーブンドレインのプルダウンによってワイヤード AND の結果がもたらされます)。

### Resume コマンド [A5h]

標準アプリケーションで、DS2408 に数回アクセスして、制御機能や調整機能を実行することができます。これは、マルチドロップ環境では Match ROM コマンドの 64 ビット ROM シーケンスをアクセスごとに繰り返す必要があるということです。マルチドロップ環境でデータスループットを最大限にするために、Resume コマンド機能が用意されています。この機能は RC フラグの状態をチェックし、それが設定されている場合は Skip ROM コマンドと同様に制御を制御機能フローに直接移します。Match ROM、Search ROM、Conditional Search ROM、または Overdrive-Match ROM コマンドを正常に実行することのみによって、RC フラグを設定することができます。RC フラグを設定すると、Resume コマンド機能によってデバイスに何度もアクセスすることができます。バス上の別のデバイスにアクセスすると、RC フラグがクリアされ、複数のデバイスが Resume コマンド機能に同時に応答するのを防止します。

### Skip ROM [3Ch]

バスマスタが 64 ビット ROM コードなしに制御機能にアクセスできるようにして、シングルドロップバス上でこのコマンドで時間を削減することができます。標準の Skip ROM コマンドとは異なり、Overdrive Skip ROM は DS2408 をオーバードライブモード(OD = 1)に設定します。継続時間が最低 480 $\mu$ s であるリセットパルスがバス上の全デバイスを標準速度(OD = 0)にリセットするまで、このコマンド発行後の通信はすべてオーバードライブ速度で行われる必要があります。このコマンドはマルチドロップバスで発行されると、オーバードライブに対応する全デバイスがオーバードライブモードに設定されます。この後で特定のオーバードライブに対応するデバイスをアドレス指定するには、Match ROM または Search ROM コマンドシーケンスが後続するオーバードライブ速度のリセットパルスを発行する必要があります。これによって検索処理時間が短縮されます。オーバードライブに対応する複数のスレーブがバス上にあり、Overdrive Skip ROM コマンドの後に Read コマンドが続く場合は、複数のスレーブが同時送信すると、データの衝突がバス上で発生します(オーブンドレインのプルダウンによってワイヤード AND の結果がもたらされます)。

### Overdrive Match ROM [69h]

オーバードライブ速度で送信される 64 ビット ROM シーケンスが後に続く Overdrive Match ROM コマンドによって、バスマスタがマルチドロップバス上の特定 DS2408 をアドレス指定し、同時にオーバードライブモードに設定することができます。64 ビット ROM シーケンスと完全一致する DS2408 のみが、以降の制御機能コマンドに応答します。このとき、前の Overdrive Skip または Match コマンドで現在オーバードライブモード状態のスレーブは、オーバードライブモードのままです。オーバードライブ対応スレーブはすべて、持続時間が最低 480 $\mu$ s の次のリセットパルスで標準速度に戻ります。バス上の 1 つまたは複数のデバイスで、Overdrive Match ROM コマンドを使用することができます。

図 12-1. ROM 機能フローチャート

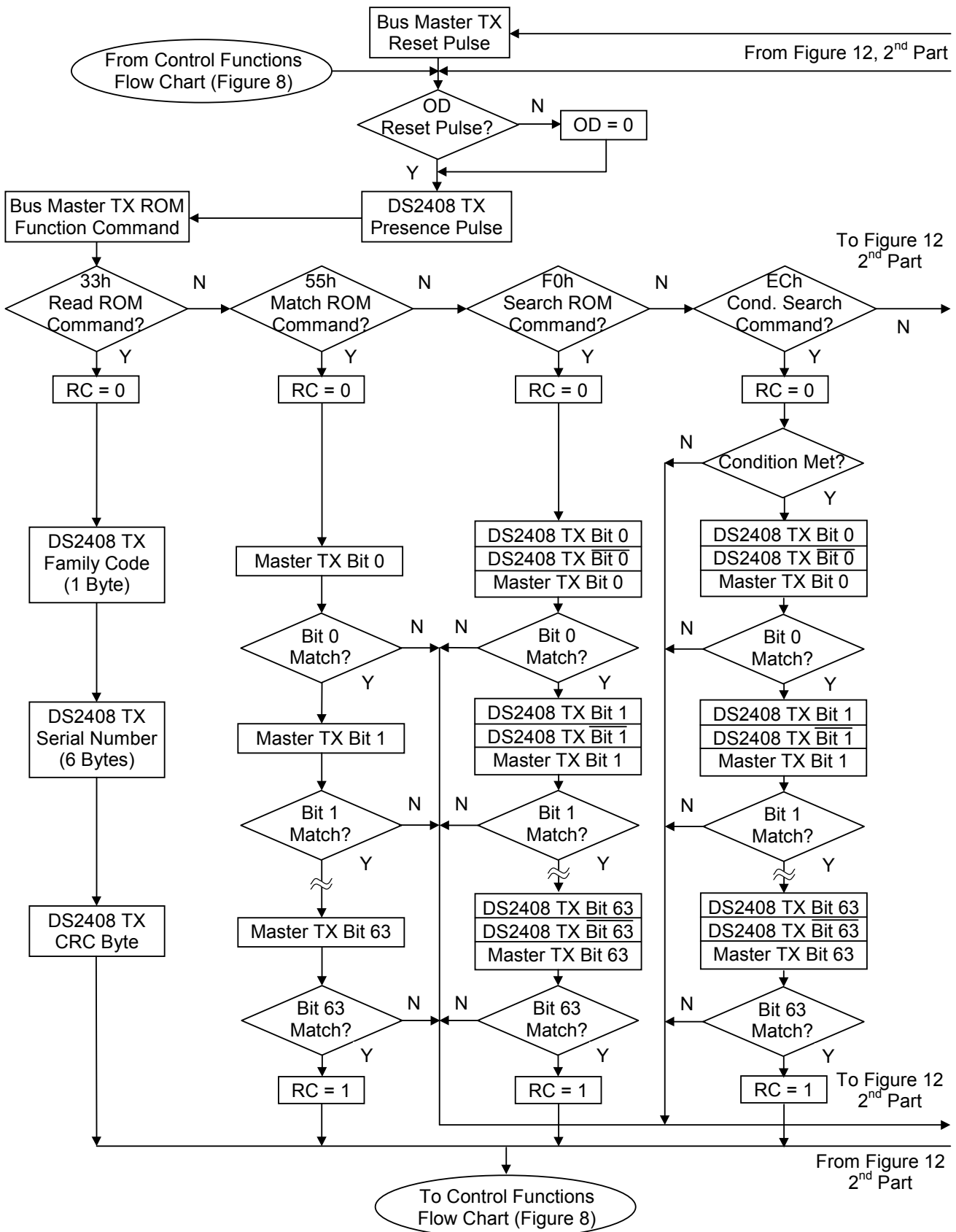
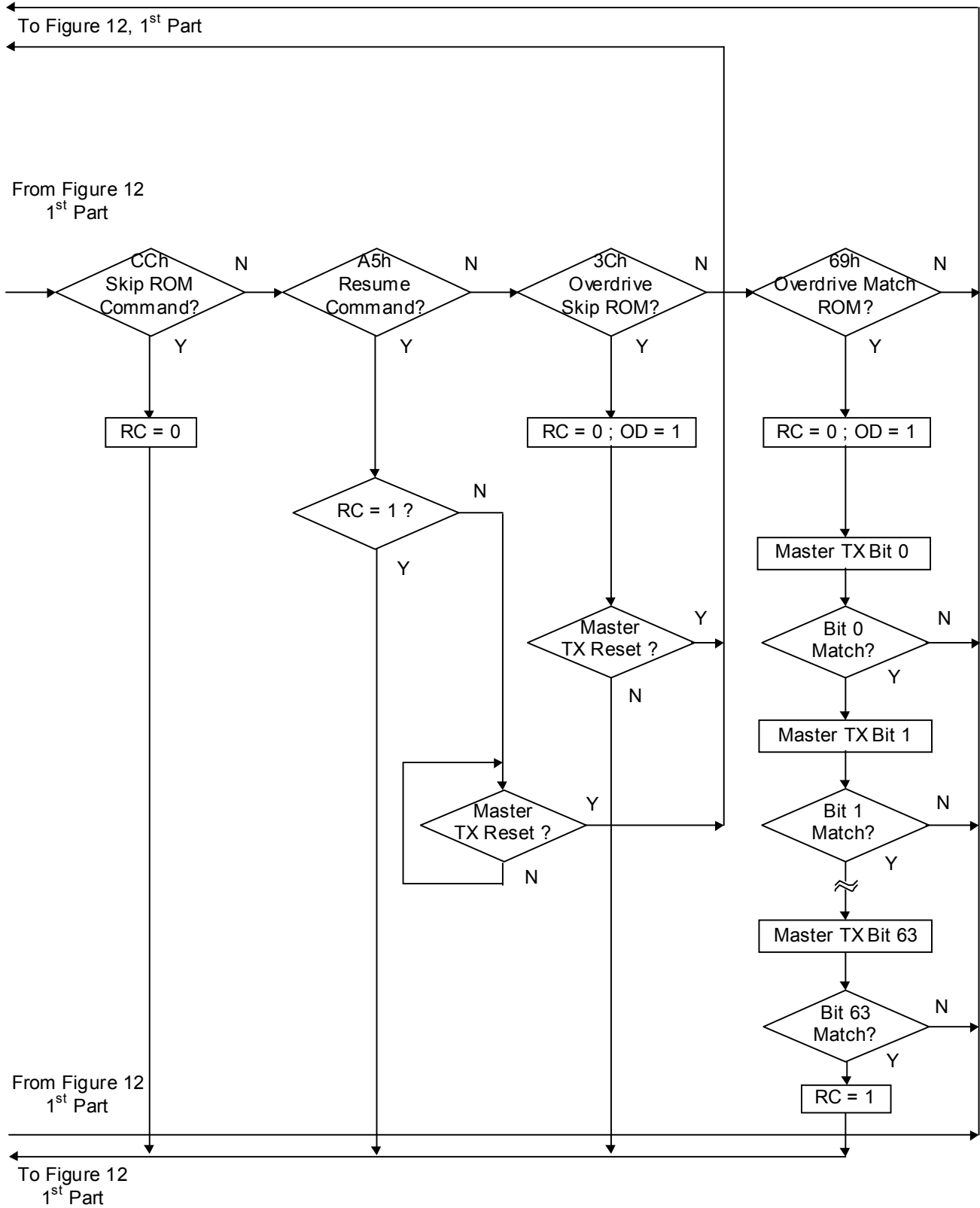


図 12-2. ROM 機能フローチャート

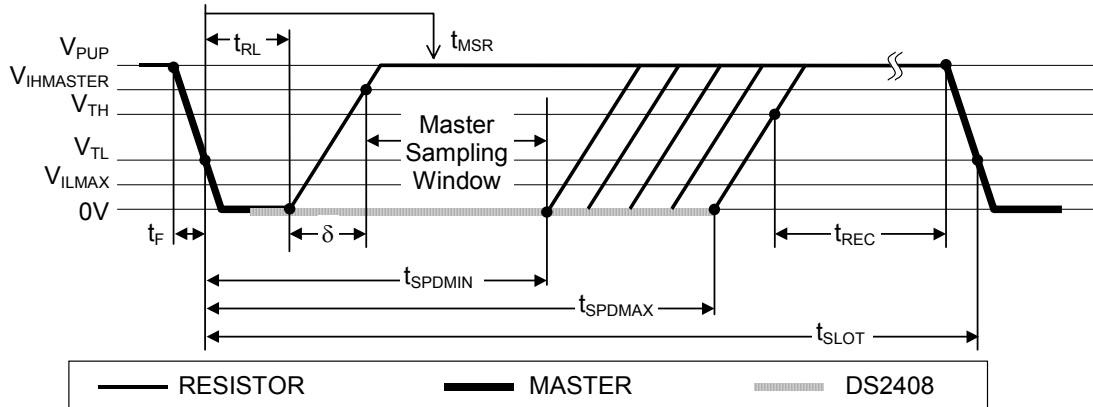








## データ読み込みタイムスロット



## スレーブからマスタに

データ読み込みタイムスロットは、Write 1 タイムスロットと同様に開始します。データラインの電圧は、読み込みロータイム  $t_{RL}$  が終了するまで、 $V_{TLMIN}$  を下回っている必要があります。 $t_{RL}$  ウィンドウの間、DS2408 は 0 で応答すると、データラインをローにプルダウンし始めます。内蔵タイミング発生器によって、このプルダウンが終了し、電圧が再度上昇を開始する時点が決定します。DS2408 が 1 で応答すると、DS2408 はデータラインをローに維持せず、 $t_{RL}$  が終了するとすぐに電圧が上昇し始めます。

一方の  $t_{RL} + \delta$  (立上り時間) と他方の DS2408 の内蔵タイミング発生器の合計によって、マスタのサンプリングウィンドウ ( $t_{MSRMIN} \sim t_{MSRMAX}$ ) が設定されます。このウィンドウ内で、マスタはデータラインから読み込みを実行する必要があります。高信頼性通信には、 $t_{RL}$  はできるだけ短くして、マスタは  $t_{MSRMAX}$  に近接して  $t_{MSRMAX}$  を超えずに読み込む必要があります。マスタはデータラインから読み込んだ後に、 $t_{SLOT}$  が終了するまで待機する必要があります。これによって、DS2408 が次のタイムスロットに備えるのに十分なりカバリ時間  $t_{REC}$  が保証されます。

## ネットワーク動作の向上

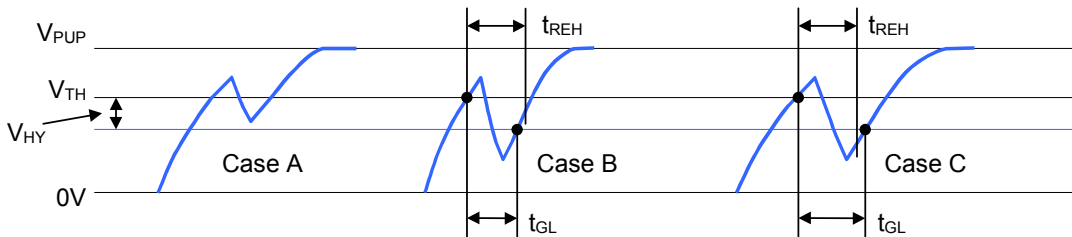
1-Wire 環境ではバスマスタ(1-Wire ドライバ)が制御するトランジエント期間のみ、ライン終端が可能です。このため、1-Wire ネットワークは各発生元からのノイズの影響を受けやすくなっています。ネットワークの物理サイズやトポロジに応じて、終端と分岐ポイントからの反射が加算され、または、ある程度相殺することができます。こうした反射は、1-Wire 通信ラインのグリッチやリングングとして見られます。また、外部ノイズ源から 1-Wire ラインに結合されたノイズによって、信号グリッチが発生する場合があります。タイムスロットの立上りエッジ時のグリッチによってスレーブデバイスがマスタとの同期を失い、このため、Search ROM コマンドが失敗したり、またデバイスレベルコマンドが異常終了します。ネットワークアプリケーションの性能を向上するために、DS2408 は新たな 1-Wire フロントエンドを使用しています。このフロントエンドはノイズへの感度を低減し、また、スレーブデバイス自体が注入するノイズの大きさも低減しています。

DS2408 の 1-Wire フロントエンドは、以下の 4 つの特性において従来のスレーブデバイスとは異なります。

- 1) プレゼンスパルスの立下りエッジは、スルーレートを制御しています。このため、デジタルでスイッチするトランジスタに比べ、ラインインピーダンスとのインピーダンス整合が向上し、従来のデバイスで知られる高周波リングングがスムーズな低周波遷移に変わっています。スルーレート制御は、パラメータ  $t_{FPD}$  で規定されます。このパラメータは標準速度とオーバードライブ速度で異なる値です。
- 2) タイムスロットの開始時の立下りエッジを検出するローパスフィルタが回路内に追加されています。これによって高周波ノイズの感度が低減しています。このフィルタは、オーバードライブ速度には適用されません。

- 3) 入力バッファにヒステリシスを設計しました。負のグリッチが  $V_{TH}$  を横切り、 $V_{TH} - V_{HY}$  間を下回らない場合は、グリッチは認識されません(図 15、ケース A)。このヒステリシスは、どの 1-Wire 速度でも有効です。
- 4) 立上りエッジのホールドオフ時間  $t_{REH}$  によって規定されるタイムウィンドウがあります。このホールドオフ時間では、グリッチがスレッシュホールド  $V_{TH} - V_{HY}$  間を下回ってもグリッチは無視されます(図 15、ケース B、 $t_{GL} < t_{REH}$ )。スレッシュホールド  $V_{TH}$  を超えた後で発生した、 $t_{REH}$  ウィンドウを超える深い電圧ドロップやグリッチは除去できず、新たなタイムスロットの開始として見なされます(図 15、ケース C、 $t_{GL} \geq t_{REH}$ )。

図 15. ノイズ抑制方式



## CRC の生成

DS2408 には、2 種類の巡回冗長検査(CRC)があります。ひとつは 8 ビットタイプで、64 ビット ROM の最上位バイトに格納されています。バスマスタは 64 ビット ROM の先頭の 56 ビットから CRC 値を算出し、DS2408 内に格納された値と照合して、ROM データがエラーなしで受信されたかを判断することができます。この CRC の等価多項式関数は、 $X^8 + X^5 + X^4 + 1$  です。この 8 ビット CRC は、true(非反転)形式で受信されます。この CRC は出荷時に算出され、ROM にレーザで書き込まれます。

もう一方の CRC は 16 ビットタイプで、標準 CRC16-多項式関数  $X^{16} + X^{15} + X^2 + 1$  にしたがって生成されます。Read PIO Registers コマンドでレジスタページの終わりまでデータを読み込むときのエラー検出、スクラッチパッドに対する書込みや読み込み時のデータ転送の迅速な検証、及び Channel-access Read コマンドで PIO から読み込むときに、この CRC が使用されます。8 ビット CRC とは対照的に、この 16 ビット CRC は常に反転形式で通信されます。DS2408 チップ内の CRC 生成器(図 16)は、図 8 のコマンドフローチャートに示されているように新たな 16 ビット CRC を算出します。バスマスタは、デバイスから読み込まれた CRC 値をデータから算出された値と照合して、動作を続行するか、または CRC エラーのデータ部分を再読み込みするかを判断します。

Read PIO Registers フローチャートの場合、16 ビット CRC 値は、まずクリア済みの CRC 生成器にコマンドバイトをシフトし、続いて 2 バイトのアドレスバイトと、ターゲットアドレスから始まり、レジスタページの最後のアドレス 008Fh で終了するデータバイトをシフトインした結果です。

Channel-access Read コマンドフローによる最初のプロセスでは、まず CRC 生成器をクリアし、コマンドコード及び後続の 32 バイトの PIO ピンデータをシフトインして、CRC が生成されます。コマンドフローによる以降のプロセスでは、CRC 生成器をクリアし、PIO ピンから読み込まれた 32 バイトをシフトインした結果である 16 ビット CRC が生成されます。CRC 値の生成の詳細については、『アプリケーションノート 27』を参照してください。

図 16. CRC-16 のハードウェア説明及び多項式

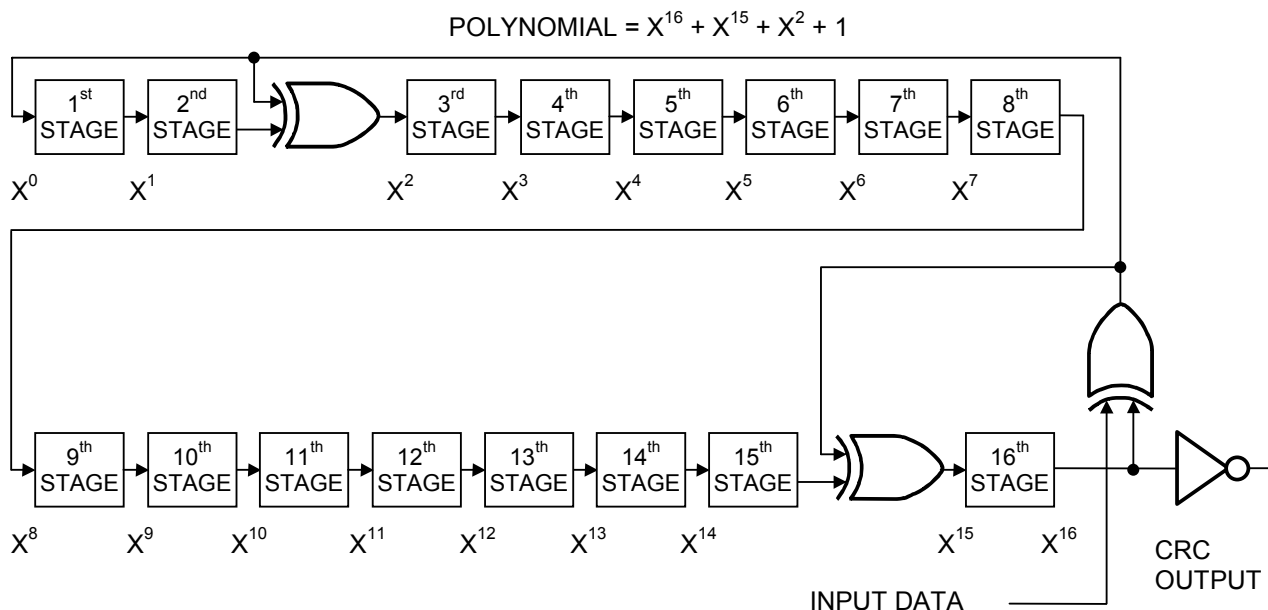
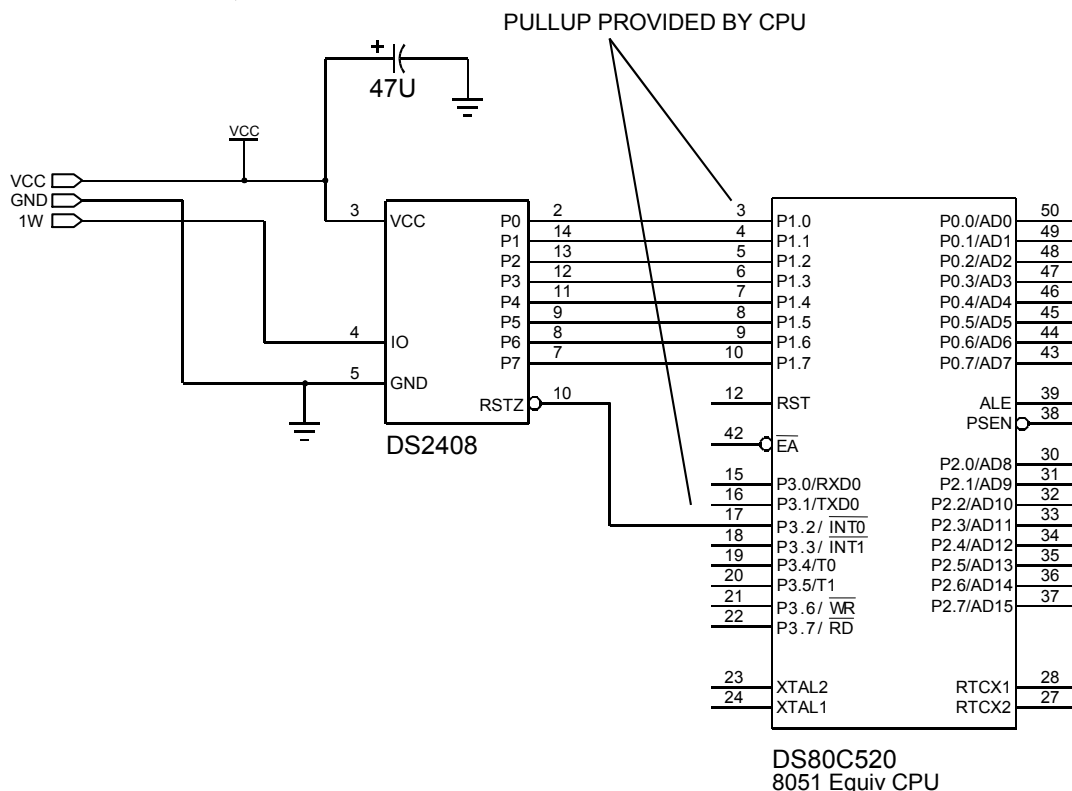


図 17. マイクロコントローラ用スレーブインタフェースとしての DS2408



データ方向(アップロード/ダウンロード)は、アプリケーション固有のデータプロトコルで決定します。

図 18. インテリジェントディスプレイ用スレーブインタフェースとしての DS2408

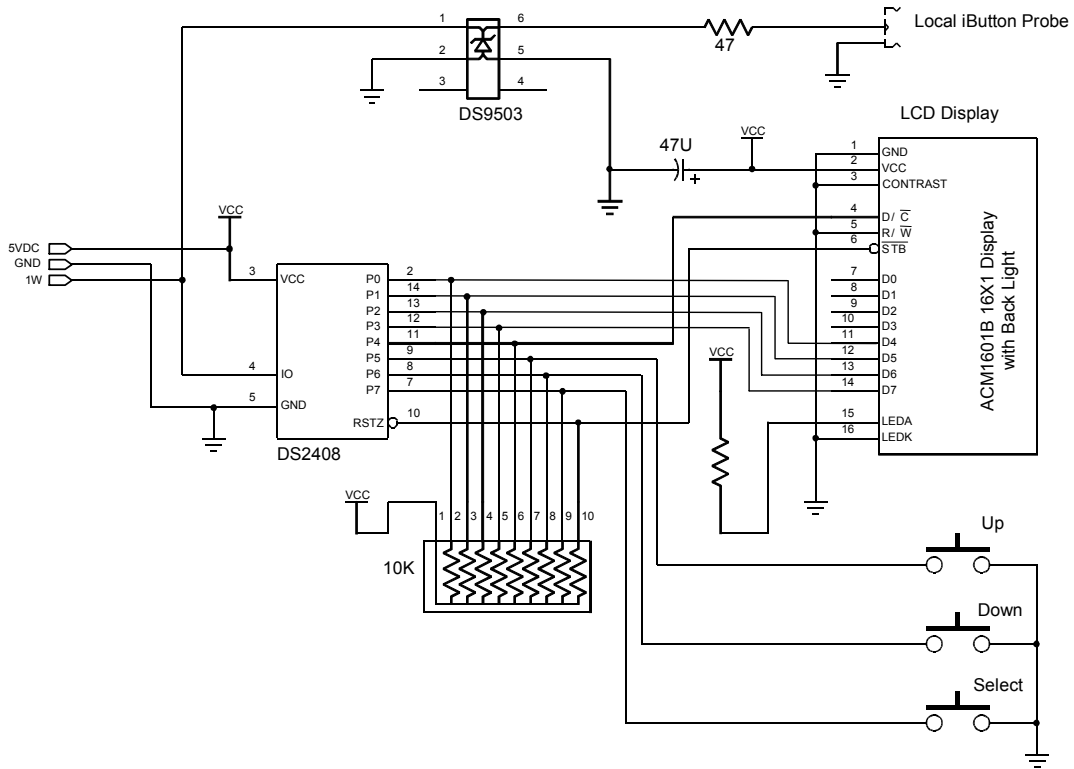
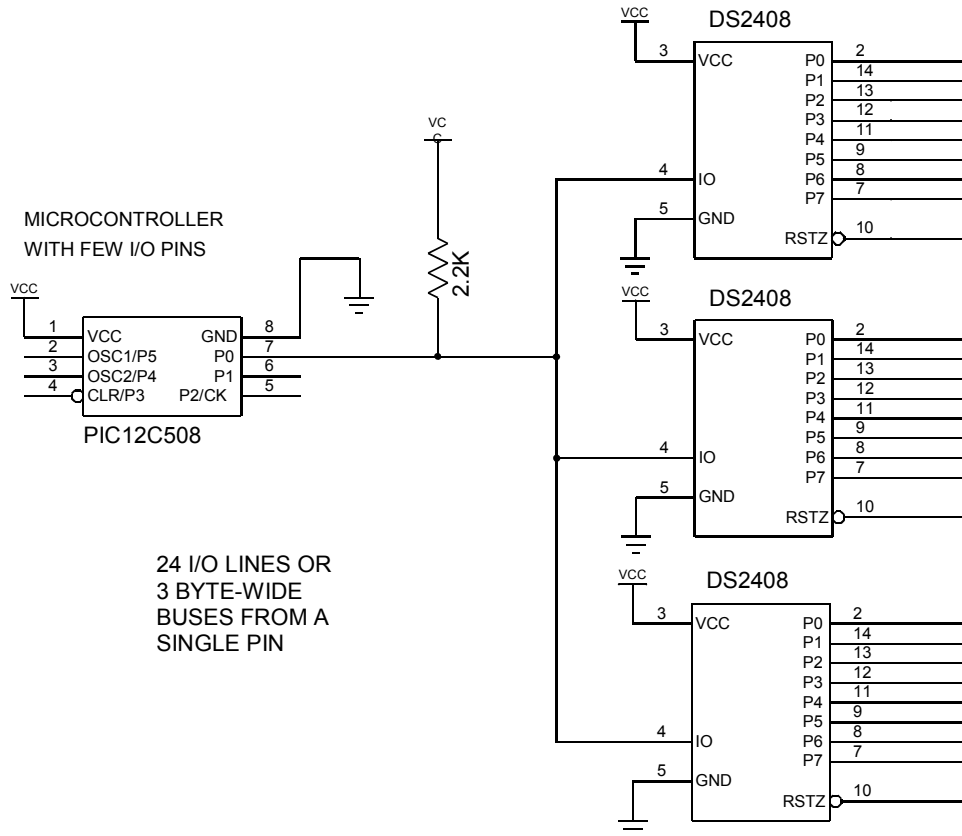


図 19. マイクロコントローラポートエキスパンダとしての DS2408



24 I/O LINES OR  
3 BYTE-WIDE  
BUSES FROM A  
SINGLE PIN

図 20. マイクロコントローラ駆動のキーボードスキャナとしての DS2408

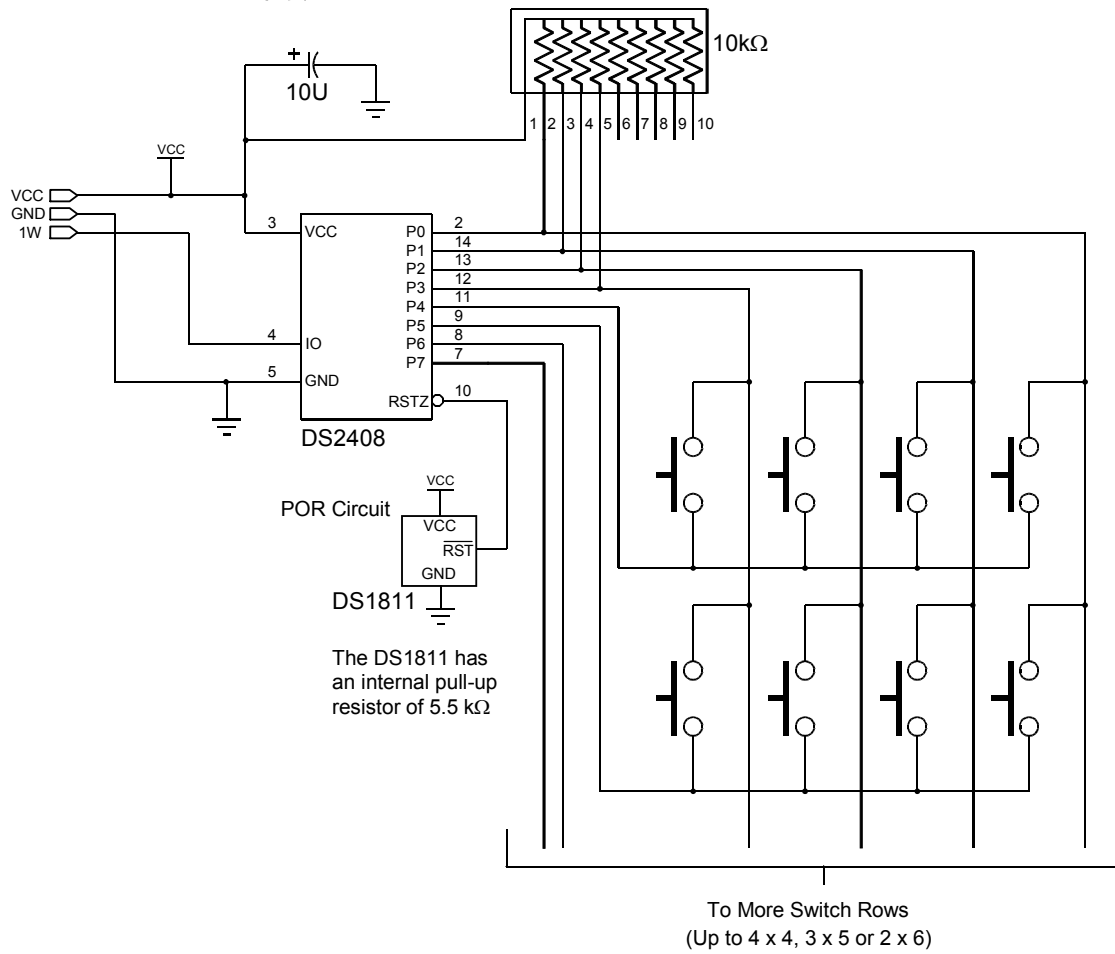


図 21. 寄生電源のプッシュボタンセンサとしての DS2408

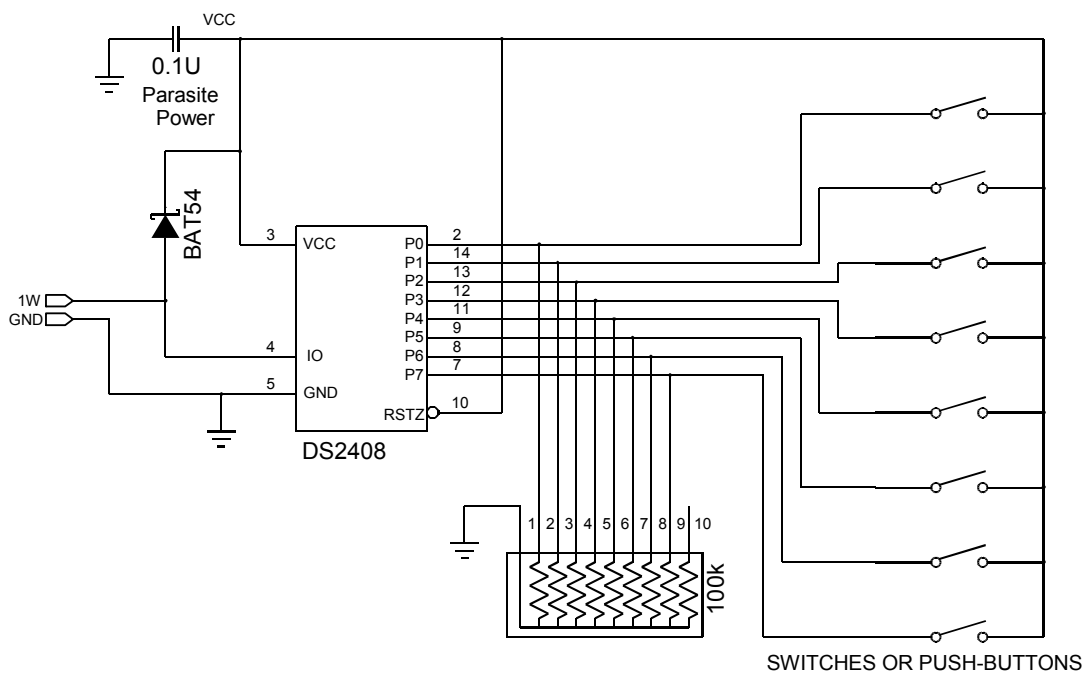
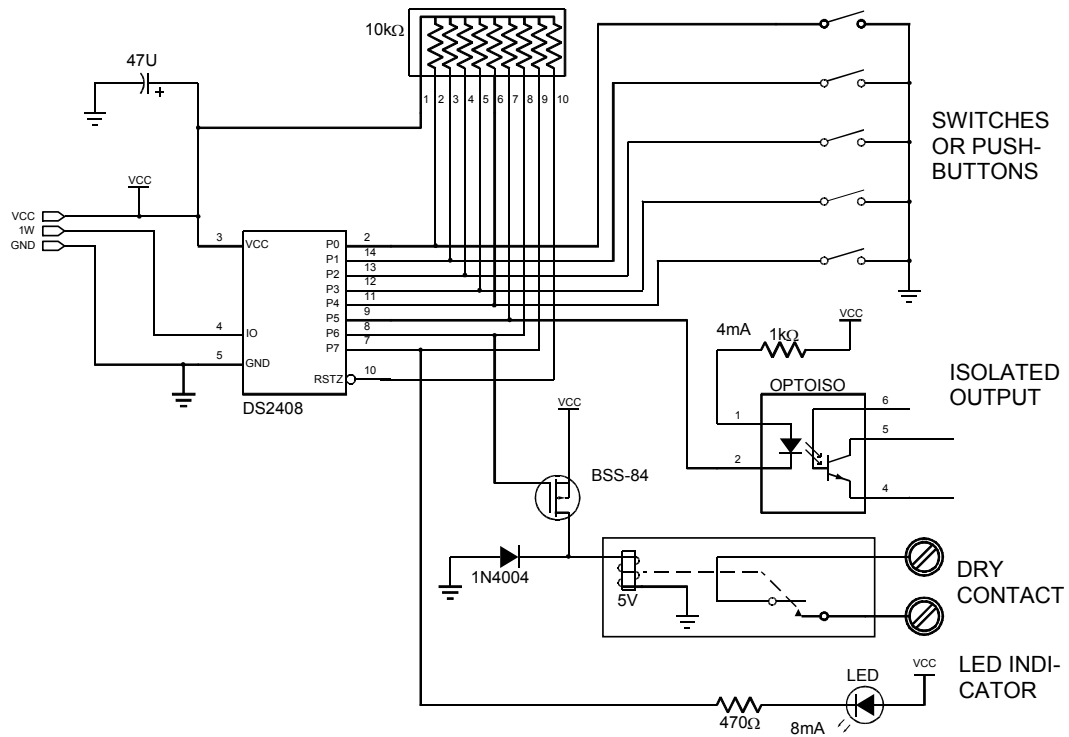


図 22. 汎用センサ/アクチュエータとしての DS2408



## コマンド固有の 1-Wire 通信プロトコル — 凡例

SYMBOL	DESCRIPTION
RST	1-Wire Reset Pulse generated by master.
PD	1-Wire Presence Pulse generated by slave.
Select	Command and data to satisfy the ROM function protocol.
RPR	Command "Read PIO Registers".
CAR	Command "Channel-Access Read".
CAW	Command "Channel-Access Write".
WCS	Command "Write Conditional Search Register".
RAL	Command "Reset Activity Latches".
TA	Target Address TA1, TA2.
<data>	Transfer of an undetermined amount of data.
CRC16\	Transfer of an inverted CRC16.
FF loop	Indefinite loop where the master reads FF bytes.
AA loop	Indefinite loop where the master reads AA bytes.
<32 samples>, CRC16\ loop	Indefinite loop where the master reads 32 PIO samples followed by an inverted CRC16.
<new state>, <new state>	Transfer of 2 bytes, where the second byte is the bit-inverse of the first byte. The first byte will be taken as the new PIO state.
AAh, <read back>	Transfer of 2 bytes, where the first byte is a constant (AAh) and the second byte is the current PIO state.
<new state>, <invalid>	Transfer of 2 bytes, where the second byte is NOT the bit-inverse of the first byte.

## コマンド固有の 1-Wire 通信プロトコル — カラーコード

Master to slave	Slave to master
-----------------	-----------------

### Read PIO Registers (成功)

RST	PD	Select	RPR	TA	<data>	CRC16\	FF loop
-----	----	--------	-----	----	--------	--------	---------

### Read PIO Registers (アドレス失敗)

RST	PD	Select	RPR	TA	FF loop
-----	----	--------	-----	----	---------

### Channel-Access Read (常時成功)

RST	PD	Select	CAR	<32 samples>, CRC16\ loop
-----	----	--------	-----	---------------------------

**Channel-Access Write (成功)**

RST	PD	Select	CAW	<new state>, <new state>	AAh, <read back>
-----	----	--------	-----	--------------------------	------------------

Loop

**Channel-Access Write (新状態失敗)**

RST	PD	Select	CAW	<new state>, <invalid>	FF loop
-----	----	--------	-----	------------------------	---------

**Write Conditional Search Register (成功)**

RST	PD	Select	WCS	TA	<data>	FF loop
-----	----	--------	-----	----	--------	---------

**Write Conditional Search Register (アドレス失敗)**

RST	PD	Select	WCS	TA	FF loop
-----	----	--------	-----	----	---------

**Reset Activity Latches (常時成功)**

RST	PD	Select	RAL	AA loop
-----	----	--------	-----	---------

**通信例**

このセクションの例は、標準状態での ROM 及び制御機能の使用を検証します。最初の 2 例は、図 17 と関連しています。この 2 例は、検証または即時応答受信用のリードバックによる PIO への書込み方法(例 1)と、永久ループでの PIO からの読み込み方法(例 2)を示しています。例 3 は、図 21 のように各デバイスが 8 個のプッシュボタンと接続している複数 DS2408 のネットワークを想定しています。

**例 1**

**タスク:** 検証または即時応答受信用のリードバックによる PIO への書込み  
このタスクは、以下のステップに分類されます。

- 1) RSTZ を STRB 出力として設定
- 2) 構成設定を検証
- 3) PIO に書込み、応答をリードバック

バスマスタと接続された単一 DS2408 のみの場合は、通信は以下の通りです。

MASTER MODE		DATA (LSB FIRST)	COMMENTS
Step 1	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	CCh	Issue Skip ROM command
	TX	CCh	Issue Write Conditional Search Register command
	TX	8Dh	TA1, target address = 8Dh
	TX	00h	TA2, target address = 008Dh
	TX	04h	Write byte to Control/Status Register



MASTER MODE		DATA (LSB FIRST)	COMMENTS
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
Step 2	TX	CCh	Issue Skip ROM command
	TX	F0h	Issue Read PIO Registers command
	TX	8Dh	TA1, target address = 8Dh
	TX	00h	TA2, target address = 008Dh
	RX	84h	Read Control/Status Register and verify
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
Step 3	TX	CCh	Issue Skip ROM command
	TX	5Ah	Issue Channel-access Write command
	TX	<PIO output byte>	Write byte to PIO
	TX	<inverted PIO output byte>	Write inverted byte to PIO
	(—)	(—)	DS2408 updates PIO status if transmission was OK
	RX	AAh	Read for verification (AAh = success)
	(—)	(—)	DS2408 samples PIO pin status
	RX	<PIO pin status byte>	Read PIO pin status
	TX	<PIO output byte>	Write byte to PIO (next byte)
	TX	<inverted PIO output byte>	Write inverted byte to PIO (next byte)
	RX	AAh	Read for verification (AAh = success)
	RX	<PIO pin status byte>	Read PIO pin status
	(—)	(—)	Repeat the previous 4 steps with more PIO output data as needed in the application.
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse

図 17 のように、この通信例を使ってデータをリモートマイクロコントローラに送信するときは、1 バイト長で始まり、CRC16 で終了するデータパケットを送信して、マスタとリモートマイクロコントローラとの同期を維持することができます。詳細については、『アプリケーションノート 114』の「UNIVERSAL DATA PACKET」の項を参照してください。

## 例 2

**タスク:** 永くループで PIO から読み込み

このタスクは、以下のステップに分類されます。

- 1) RSTZ を STRB 出力として設定
- 2) 構成設定を検証
- 3) PIO から読み込み

バスマスタと接続された単一 DS2408 のみの場合は、通信は以下の通りです。

MASTER MODE		DATA (LSB FIRST)	COMMENTS
Step 1	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	CCh	Issue Skip ROM command
	TX	CCh	Issue Write Conditional Search Register command
	TX	8Dh	TA1, target address = 8Dh
	TX	00h	TA2, target address = 008Dh
	TX	04h	Write byte to Control/Status Register

MASTER MODE	DATA (LSB FIRST)	COMMENTS
TX	(Reset)	Reset pulse
RX	(Presence)	Presence pulse
Step 2 TX	CCh	Issue Skip ROM command
TX	F0h	Issue Read PIO Registers command
TX	8Dh	TA1, target address = 8Dh
TX	00h	TA2, target address = 008Dh
RX	84h	Read Control/Status Register and verify
TX	(Reset)	Reset pulse
RX	(Presence)	Presence pulse
Step 3 TX	CCh	Issue Skip ROM command
TX	F5h	Issue Channel-access Read command
(—)	(—)	DS2408 samples PIO pin status
RX	<PIO pin status byte>	Read PIO pin status
(—)	(—)	Repeat the previous 2 steps until the master has received a total of 32 bytes of PIO pin status
RX	<2 bytes CRC16>	Read CRC16
(—)	(—)	PIO pin status and CRC loop can be continued as long as the application requires.
TX	(Reset)	Reset pulse
RX	(Presence)	Presence pulse

図 17 のように、この通信例を使ってデータをリモートマイクロコントローラから読み込むときは、1 バイト長で始まり、CRC16 で終了するデータパケットを送信して、マスタリモートマイクロコントローラとの同期を維持することができます。詳細については、『アプリケーションノート 114』の「UNIVERSAL DATA PACKET」の項を参照してください。

### 例 3

**タスク:** ボタンが押された特定の DS2408 を検出し、プッシュボタンが接続されている端子を識別します。このタスクは、以下のステップに分類されます。

- 1) 条件検索を設定し、構成設定を検証
- 2) すべてのチャンネル出力トランジスタをオフ
- 3) アクティビティラッチをクリア
- 4) プッシュボタンが押されるまで検索
- 5) デバイスとプッシュボタンを識別し、アクティビティラッチをリセット

8 チャンネルのうち最低 1 つのアクティビティラッチが設定されている場合は、デバイスは条件検索に応答する必要があります。このため、条件検索レジスタには以下の設定データが必要です。

**Channel Selection Mask(チャンネル選択マスク)**、全チャンネルを選択 ⇒ FFh

**Channel Polarity Selection(チャンネル極性選択)**、全チャンネルに対してロジック 1 を選択 ⇒ FFh

**制御/状態レジスタ、**

ソースはアクティビティラッチ ⇒ PLS = 1

項目は OR ⇒ CT = 0

RSTZ = RST(入力) ⇒ ROS = 0

パワーオンリセットラッチをクリア ⇒ PORL = 0

上記の結果、制御/状態レジスタの設定データは、01h です。

このアプリケーションでは各 DS2408 に、以下の初期化を実行します。

MASTER MODE		DATA (LSB FIRST)	COMMENTS
Step 1	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	55h	Issue Match ROM command
	TX	<8 byte ROM ID>	Send ROM ID of the device to be accessed
	TX	CCh	Issue Write Conditional Search Register command
	TX	8Bh	TA1, target address = 8Bh
	TX	00h	TA2, target address = 008Bh
	TX	FFh	Write Channel Selection Mask
	TX	FFh	Write Channel Polarity Selection
	TX	01h	Write Control/Status Register
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	A5h	Issue Resume command
	TX	F0h	Issue Read PIO Registers command
	TX	8Bh	TA1, target address = 8Bh
	TX	00h	TA2, target address = 008Bh
	RX	<FFh, FFh, 81h>	Read Registers and verify
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
Step 2	TX	A5h	Issue Resume command
	TX	5Ah	Issue Channel-access Write command
	TX	FFh	Write byte to PIO
	TX	00h	Write inverted byte to PIO
	(—)	(—)	DS2408 switches off all channel output transistors if transmission was OK
	RX	AAh	Read for verification (AAh = success)
	RX	FFh	Read PIO pin status and verify; FFh = OK
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
Step 3	TX	A5h	Issue Resume command
	TX	C3	Issue Reset Activity Latch command
	RX	AAh	Read for verification (AAh = success)
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse

すべての DS2408 が初期化された後に、永くループとして以下の検索プロセスを実行します。

MASTER MODE		DATA (LSB FIRST)	COMMENTS
Step 4	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	ECh	Issue Conditional Search ROM command
	RX	<2 bits>	Read 2 bits; if both bits are 1, no push button has been pressed; in this case return to Step 4. If the bit pattern is 01 or 10 or 00, a push button has been pressed; in this case continue with Step 5.
Step 5	TX	<1 bits>	Identify and select the LS bit of the ROM ID of the DS2408 that has responded to the Conditional Search.
	RX	<2 bits>	Read 2 bits; this relates to the next bit of the ROM ID of the participating device(s).
	TX	<1 bits>	Identify and select the next bit of the ROM ID of the DS2408 that has responded to the Conditional Search.
	(—)	(—)	Repeat the previous 2 steps until one device has been identified and accessed. (see Note 1)
	TX	F0h	Issue Read PIO Registers command
	TX	88h	TA1, target address = 88h
	TX	00h	TA2, target address = 0000h
	RX	<8 data bytes>	Read register page; the data in the Activity Latch State Register tells which button has been pressed.
	RX	<2 bytes CRC16>	Read CRC16 and verify correct data transmission.
	TX	(Reset)	Reset pulse
	RX	(Presence)	Presence pulse
	TX	A5h	Issue Resume command
	TX	C3	Issue Reset Activity Latch command
	RX	AAh	Read for verification (AAh = success)
	(—)	(—)	Now, as the device and push button are identified and the Activity Latch is cleared, continue at Step 4.

**注記 1:** 検索アルゴリズムの詳細については、『アプリケーションノート 187』を参照してください。