

具有实时时钟功能的 1-Mbit (128 K × 8)  
串行 (I<sup>2</sup>C) nvSRAM

## 特性

### ■ 1 Mbit 非易失性静态随机存取存储器 (nvSRAM)

- 内部采用 128K x 8 的组织方式
- 在断电时自动启动（自动存储）或者由用户使用 I<sup>2</sup>C 命令（软件存储）或 HSB 引脚（硬件存储）启动数据存储至 QuantumTrap 非易失性组件
- 在通电时启动（加电回读）或者由 I<sup>2</sup>C 命令（软件回读）启动数据回读至 SRAM
- 只需一个小电容，即可在断电时实现自动存储

■ 可靠性高

- 无限次读、写和回读循环
- 一百万次 QuantumTrap 存储循环
- 数据保留时间：20 年 (85 °C)

## ■ 实时时钟 (RTC)

- 功能齐全的实时时钟
- 看门狗定时器
- 带可编程中断的时钟警报
- 备用电源失败指示
- 可编程频率（1 Hz、512 Hz、4096 Hz 和 32.768 kHz）方波输出
- RTC 备用电容或电池
- 备用电流为 0.45  $\mu$ A（典型值）

## ■ 高速 I<sup>2</sup>C 接口<sup>[1]</sup>

- 行业标准 100 kHz 和 400 kHz 速度
- 增强型快速模式: 1 MHz 速度
- 高速: 3.4 MHz
- 零循环延迟读写

## ■ 写保护

- 使用写保护 (WP) 引脚提供硬件保护
- 可为 1/4、1/2 或整个阵列提供软件模块保护

## ■对特殊功能的 I<sup>2</sup>C 访问

- ❑ 非易失性存储 / 回读
- ❑ 8 字节序列号
- ❑ 制造商 ID 和产品 ID
- ❑ 睡眠模式

### ■ 低功耗

- 工作频率为 3.4 MHz 时, 平均有功电流为 1 mA
- 待机模式下的平均电流为 250  $\mu$ A
- 睡眠模式下的电流为 8  $\mu$ A

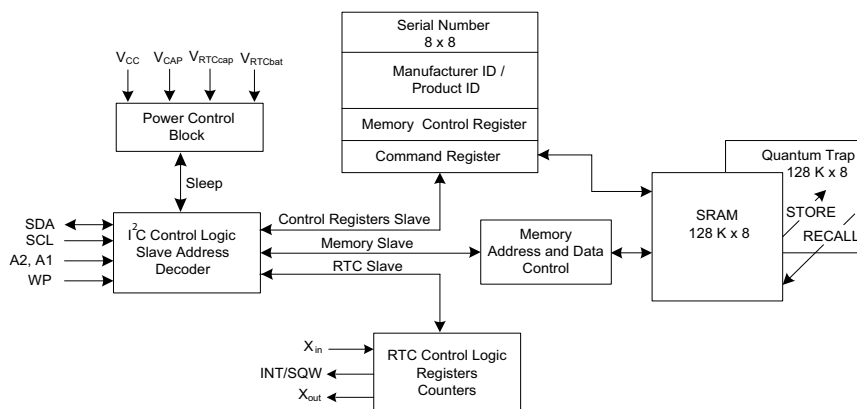
### ■ 行业标准配置

- 工作电压:
  - CY14C101I:  $V_{CC} = 2.4\text{ V}$  至  $2.6\text{ V}$
  - CY14B101I:  $V_{CC} = 2.7\text{ V}$  至  $3.6\text{ V}$
  - CY14E101I:  $V_{CC} = 4.5\text{ V}$  至  $5.5\text{ V}$
- 工业级温度
- 16 引脚小外形集成电路 (SOIC) 封装
- 无铅并符合有害物质限制 (RoHS)

## 概述

赛普拉斯 CY14C101I/CY14B101I/CY14E101I 将 1 Mbit 的 nvSRAM<sup>[2]</sup> 和功能齐全的实时时钟通过串行 I<sup>2</sup>C 接口整合在了一个单片集成电路中。该存储器采用“128 K 字, 每字 8 位”的组织方式。嵌入式非易失性元件通过采用 QuantumTrap 技术, 打造出了世界上最可靠的非易失性存储器。SRAM 能够实现无限次读写循环, 而 QuantumTrap 单元则能够提供高度可靠的非易失性数据存储空间。断电时, 数据会从 SRAM 自动转移到非易失性元件中 (存储操作)。加电时, 数据会从非易失性存储器回读到 SRAM (回读操作)。存储和回读操作也可以由用户通过 I<sup>2</sup>C 命

### 逻辑框图



注

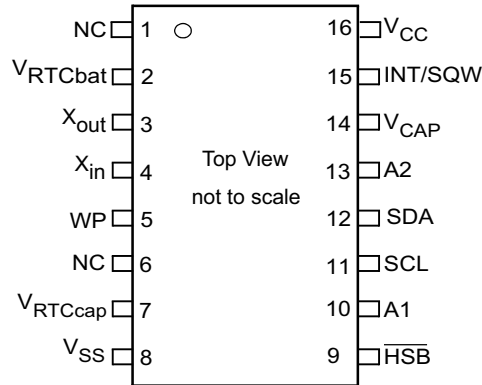
1.  $I^2C$  nvSRAM 在以下范围内不符合 NXP  $I^2C$  规范:  $V_{OL}/I_{OL}$ 、I/O 迟滞和  $t_{tOF}$  规范。有关详细信息,请参见[直流电气特性](#)和[交流开关特性表](#)。
2. 在整个数据手册中,都将串行 ( $I^2C$ ) nvSRAM 称作 nvSRAM。

## 目录

引脚分布 .....	3	看门狗定时器 .....	21
引脚定义 .....	3	可编程方波发生器 .....	22
I2C 接口 .....	4	功耗监控器 .....	22
协议概述 .....	4	后备电源监控器 .....	22
I2C 协议—数据传输 .....	4	中断 .....	22
数据有效性 .....	5	中断寄存器 .....	22
START 条件 (S) .....	5	标志寄存器 .....	23
STOP 条件 (P) .....	5	<b>最大额定值 .....</b>	<b>29</b>
重复 START (Sr) .....	5	工作范围 .....	29
字节格式 .....	5	直流电气特性 .....	29
确认 / 未确认 .....	5	数据保留与耐久性 .....	30
高速模式 (Hs 模式) .....	6	热阻 .....	30
从组件地址 .....	7	交流测试负载和波形 .....	31
写保护 (WP) .....	9	交流测试条件 .....	31
自动存储操作 .....	9	RTC 特性 .....	31
硬件存储和 HSB 引脚操作 .....	9	交流开关特性 .....	32
硬件回读 (加电) .....	10	开关波形 .....	32
写操作 .....	10	nvSRAM 规格 .....	33
读操作 .....	10	开关波形 .....	33
存储器从组件访问 .....	10	软件控制的存储 / 回读循环 .....	34
RTC 寄存器从组件访问 .....	14	开关波形 .....	34
控制寄存器从组件 .....	16	硬件存储循环 .....	35
<b>序列号 .....</b>	<b>18</b>	开关波形 .....	35
序列号写入 .....	18	订购信息 .....	36
序列号锁定 .....	18	订购代码定义 .....	36
序列号读取 .....	18	<b>封装图 .....</b>	<b>37</b>
<b>组件 ID .....</b>	<b>19</b>	<b>缩略语 .....</b>	<b>38</b>
使用命令寄存器执行命令 .....	19	<b>文档规范 .....</b>	<b>38</b>
<b>实时时钟操作 .....</b>	<b>20</b>	测量单位 .....	38
nvTIME 操作 .....	20	<b>文档修订记录页 .....</b>	<b>39</b>
时钟操作 .....	20	<b>销售、解决方案和法律信息 .....</b>	<b>40</b>
读取时钟 .....	20	全球销售和设计支持 .....	40
设置时钟 .....	20	产品 .....	40
备用电源 .....	20	PSoC 解决方案 .....	40
停止和启动振荡器 .....	20		
校准时钟 .....	21		
警报 .....	21		

## 引脚分布

图 1. 16 引脚 SOIC 的引脚分布



## 引脚定义

引脚名称	I/O 类型	说明
SCL	输入	时钟：以 $f_{SCL}$ 的最大速度运行
SDA	输入 / 输出	I/O：通过 I <sup>2</sup> C 接口输入和输出数据。 输出：为开漏并需要外部上拉电阻。
WP	输入	写保护：防止对存储器进行任何写入。引脚在内部被置于低电平状态，因此，如果未连接，则可保持为开路。
A2-A1	输入	从组件地址：为 I <sup>2</sup> C 定义从组件地址。这些引脚在内部被置于低电平状态，因此，如果未连接，则可保持为开路。
$\overline{HSB}$	输入 / 输出	硬件存储繁忙： 输出：低电平时指示 nvSRAM 繁忙状态。在每次硬件和软件存储操作之后， $\overline{HSB}$ 通过标准输出高电流变为高电平一小段时间 ( $t_{HHHD}$ )，然后通过内部弱上拉电阻一直保持高电平（外部上拉电阻连接可选）。 输入：通过从外部将引脚置于低电平状态实现硬件存储。
V <sub>CAP</sub>	电源	自动存储电容：在将数据从 SRAM 存储到非易失性元件的过程中断电时为 nvSRAM 提供电源。如不需要可禁用自动存储功能，该引脚保持未连接。该引脚绝对不可以接地。
V <sub>RTCcap</sub> <sup>[3]</sup>	电源	RTC 备用电容：如果要使用 V <sub>RTCbat</sub> ，则必须让其保持未连接状态。
V <sub>RTCbat</sub> <sup>[3]</sup>	电源	RTC 备用电池：如果要使用 V <sub>RTCcap</sub> ，则必须让其保持未连接状态。
X <sub>out</sub> <sup>[3]</sup>	输出	晶振输出连接
X <sub>in</sub> <sup>[3]</sup>	输入	晶振输入连接
INT/SQW <sup>[3]</sup>	输出	中断输出 / 校准 / 方波。可编程以响应时钟警报、看门狗定时器和功耗监控器 还可通过编程设置为高电平（推或拉）或低电平（开漏）有效。在校准模式下，可输出 512 Hz 方波。在方波模式下，用户可选择 1 Hz、512 Hz、4096 Hz 或 32768 Hz 频率做为连续输出。
NC	无连接	无连接。该引脚未连接到 die。
V <sub>SS</sub>	电源	接地
V <sub>CC</sub>	电源	电源

### 注

3. 如果不使用 RTC 功能，则必须让其保持未连接状态。

## I<sup>2</sup>C 接口

I<sup>2</sup>C 总线由两条线路组成 — 串行时钟线 (SCL) 和串行数据访问 (SDA)，可在总线上多个组件间传递信息。I<sup>2</sup>C 支持多主控和多从配置。数据在 SDA 线上从发射器传输至接收器，并与主组件生成的时钟 SCL 同步。

SCL 和 SDA 线为开漏线，并通过寄存器上拉至 V<sub>CC</sub>。系统中对上拉电阻的选择取决于总线电容及期望的运行速度。主组件生成时钟，通过该时钟所有数据 I/O 可实现同步传输。在 SCL 线上，CY14X101I 支持最大 3.4 MHz 的时钟速度。

## 协议概述

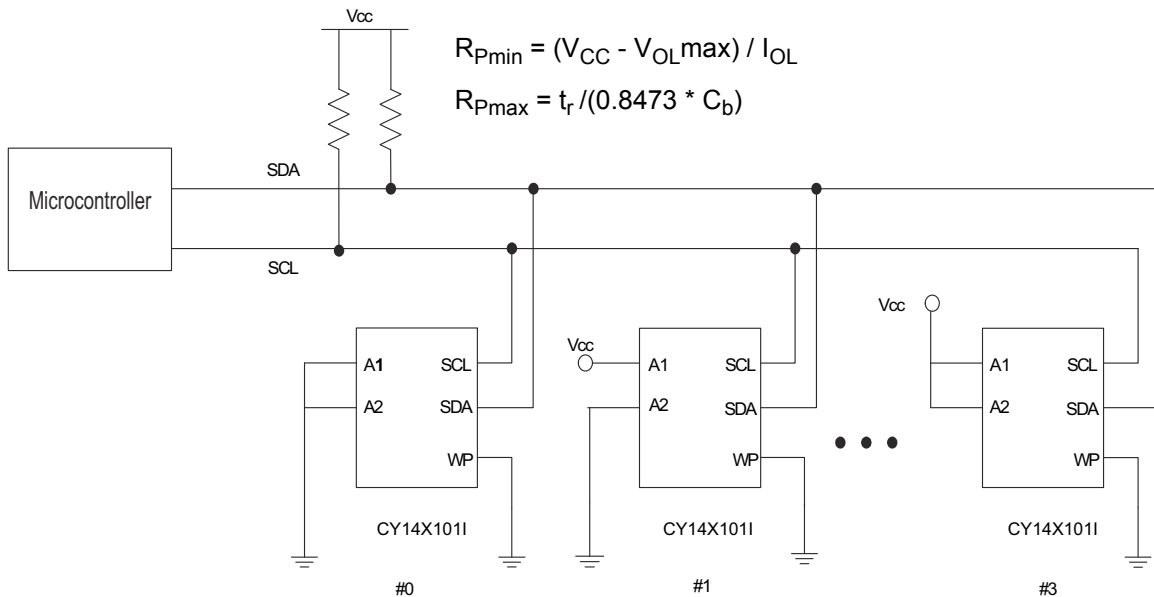
该组件仅支持 7 位寻址方案。主组件通过生成 START 条件来启动通信，并随后传播从组件选择字节。从组件选择字节由主组件准备与之建立通信的从组件地址（7 位）和一个指示读或写操作的 R/W 位组成。所选的从组件发出确认 (ACK) 信息进行响应。选择从组件后，通信的剩余部分在主组件与所选从组件间进行。总线上的其他组件忽略 SDA 线上的信号，直到检测到 STOP 或重复 START 条件。数据通过 SDA 引脚在主组件和所选从组件间传输。SDA 引脚与主组件生成的 SCL 时钟同步。

## I<sup>2</sup>C 协议—数据传输

每次通过 I<sup>2</sup>C 协议进行数据操作时，主组件首先在总线生成 START 条件，后跟 7 位从组件地址和指示读 (1) 或写 (0) 操作的第 8 位 (R/W)。所有信号在开漏 SDA 线传输并与 SCL 线上的时钟同步。通过使 SDA 线在第九个时钟脉冲保持低电平状态，接收器确认在 I<sup>2</sup>C 总线上传输的数据的每字节。主组件发出的写入请求后跟存储器地址和 SDA 线上的数据字节。可通过发送多字节数据在突发模式下进行写入操作。在第九个时钟循环的下降沿发送和接收每字节后，存储器地址自动递增。新地址在发送和接收确认位前被锁存。这样就可以访问下一连续字节而无需额外寻址。到达最后一个存储器位置时，地址返回至 0x00000，写入继续。在进行写操作期间，从组件使用 ACK 对主组件发送的每个字节做出响应。主组件可通过生成 STOP 或重复 START 条件终止写序列。

在当前地址位置（上次读或写访问位置的下一个地址）请求读取。存储器从组件通过当前地址位置向主组件发送数据对读请求做出响应。可向预备读取地址发送写请求进行随机地址读取。主组件必须在最后地址字节后立即中止写操作，并发出重复 START 或 STOP 信号以阻止任何写操作。后续的读操作将从该地址进行。主组件通过使 SDA 引脚在第九个时钟脉冲保持低电平状态来确认接收到一个字节的的数据。主组件可通过在最后数据字节后在 SDA 线发送未确认 (NACK) 信号终止读取。NACK 信号导致 CY14X101I 释放 SDA 线，然后主组件通过生成 STOP 或重复 START 条件来启动新操作。

图 2. 采用串行 (I<sup>2</sup>C) nvSRAM 进行系统配置



## 数据有效性

在时钟的高电平周期内，SDA 线上的数据必须保持稳定。只有在 SCL 线上的时钟被置于低电平以使数据有效时，才可改变数据线路的状态。当 SCL 线保持高电平时，在两种情况下可更改 SDA 线的状态。START 和 STOP 条件。主组件通过生成 START 和 STOP 条件来表示 I<sup>2</sup>C 总线上通信序列的开始和结束。

## START 条件 (S)

SCL 为高电平时 SDA 线上由高电平到低电平的切换表示 START 条件。在 I<sup>2</sup>C 中的每次数据操作过程中，主组件首先生成 START 条件。

## STOP 条件 (P)

SCL 为高电平时 SDA 线上由低电平到高电平的切换表示 STOP 条件。该条件表示结束正在进行的数据操作。

START 和 STOP 条件始终由主组件生成。在生成 START 条件后，总线被视为繁忙。在生成 STOP 条件后，总线被再次视为空闲。

## 重复 START (Sr)

如果主组件生成重复 START 条件而非 STOP 条件，总线保持繁忙状态。I<sup>2</sup>C 线上的数据操作被停止，总线等待主组件发送从组件 ID 来确认是否重新启动。

图 3. START 和 STOP 条件

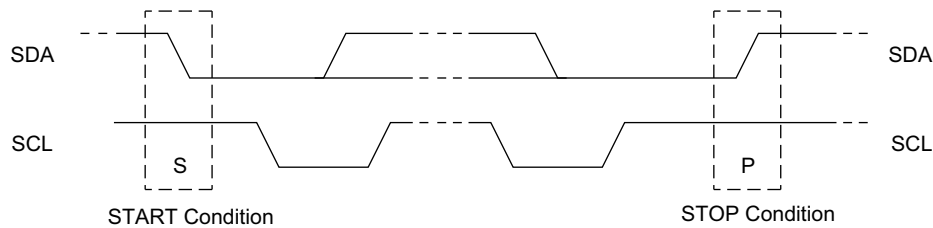
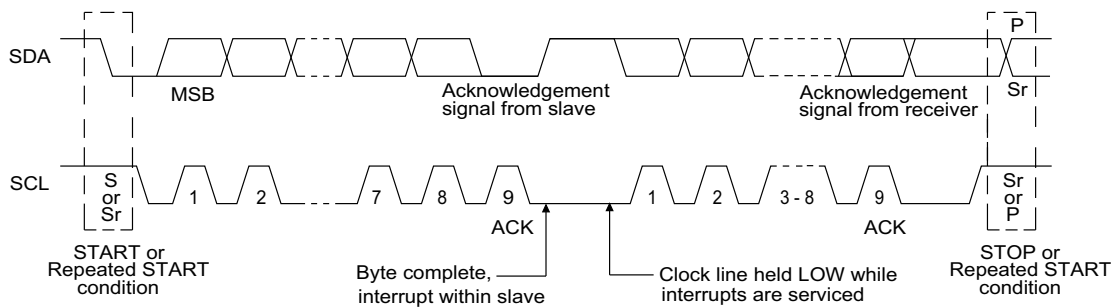


图 4. I<sup>2</sup>C 总线上的数据传输



## 字节格式

I<sup>2</sup>C 中的所有操作均采用 8 位字。在 SDA 线上按照最高有效位优先格式发送位，每字节后跟接收器发送的 ACK 信号。

操作一直持续到接收器发送 NACK 信号或主组件生成 STOP 或重复 START 条件。在时钟 (SCL) 被置于高电平状态时，除非主组件生成 START 或 STOP 条件，SDA 线必须保持稳定。

## 确认 / 未确认

发送一个字节的数据或地址后，发射器释放 SDA 线。接收器通过将 SDA 线置于低电平确认收到字节。在 I<sup>2</sup>C 总线上传输的每各字节数据需要接收器发送 ACK 信号响应以保持操作继续。未收到 ACK 信号将被视为 NACK 状态。NACK 状态下，接收器否认收到数据，操作将被中止。

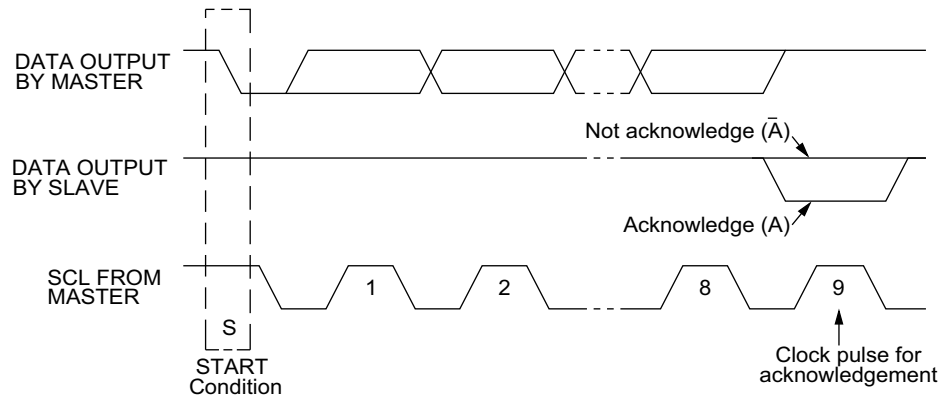
在以下情况下，主组件会在读操作期间生成 NACK 信号：

- 由于噪声，主组件未收到有效数据。
- 主组件通过生成 NACK 中止读序列。主组件发出 NACK 后，nvSRAM 从组件释放对 SDA 引脚的控制，主组件可自由生成重复 START 或 STOP 条件。

在以下情况下，nvSRAM 从组件会在写操作期间生成 NACK 信号：

- 由于噪声，nvSRAM 未收到有效数据。
- 主组件尝试在 nvSRAM 上访问写保护区。主组件必须通过生成 STOP 或重复 START 条件重新启动通信。

图 5. 在 I<sup>2</sup>C 总线上确认



### 高速模式 (Hs 模式)

在 Hs 模式下, nvSRAM 可以最大 3.4 Mbit/s 的比特率传输数据。必须发出主组件代码 (0000 1XXXb) 将组件置于高速模式。这使主组件和从组件之间以最大 3.4 MHz 的速度进行通信。停止条件将退出 Hs 模式。

#### Hs 模式下串行数据格式

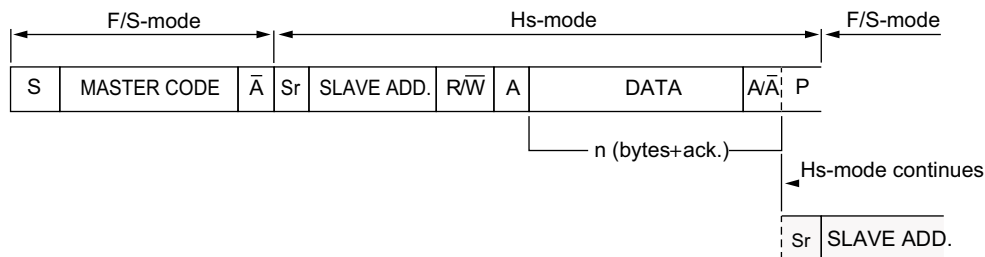
Hs 模式下串行数据传输格式符合标准模式 I<sup>2</sup>C 总线规范。Hs 模式仅在满足以下条件后开始 (所有条件均处于 F/S 模式):

1. START 条件 (S)
2. 8 位主组件代码 (0000 1XXXb)
3. 未确认位 ( $\bar{A}$ )

支持单字节和多字节读写。组件进入 Hs 模式后, 数据继续在 Hs 模式上传输, 直到主组件发送停止条件。从组件在 STOP 条件 (P) 后重新切换到 F/S 模式。要继续在 Hs 模式下传输数据, 主组件发送重复 START (Sr)。

有关针对读写操作的 Hs 模式时序, 请参见第 11 页上的图 13 和第 12 页上的图 16。

图 6. Hs 模式下的数据传输格式





## 从组件地址

I<sup>2</sup>C 总线上的每个从组件都有一个组件选择地址。START 条件后首字节包含主组件准备与之建立通信的从组件地址。七个最高有效位为组件地址，最低有效位（R/W 位）用于表示读或写操作。CY14X101I 在从组件的地址字段中保留三组前四个最高有效位 [7:4]，以便访问存储器、RTC 寄存器和控制寄存器。访问机制将在下一节中介绍。

**表 1. 从组件寻址**

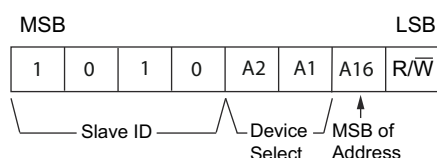
位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	nvSRAM 功能选择	
1	0	1	0	组件选择 ID		A16	R $\overline{W}$	选择存储器	<b>CY14X101I 从组件</b> <div>存储器, 128 K × 8</div> <div>RTC 寄存器, 16 × 8</div> <div>           控制寄存器            - 存储器控制寄存器, 1 × 8            - 序列号, 8 × 8            - 组件 ID, 4 × 8            - 命令寄存器, 1 × 8         </div>
1	1	0	1	组件选择 ID		X	R $\overline{W}$	选择 RTC 寄存器	
0	0	1	1	组件选择 ID		X	R $\overline{W}$	选择控制寄存器	

### 存储器从组件

如果主组件发出的从组件地址为 1010b 后跟两位组件选择，则选择 nvSRAM 组件进行读 / 写操作。如果主组件发出的从组件地址与存储器从组件地址相匹配，则根据从组件地址的 R/W 位，将从 nvSRAM 读取数据（R/W = “1”）或向 nvSRAM 写入数据（R/W = “0”）。

CY14X101I 的地址长度为 17 位，因此需要三个地址字节来映射整个存储器地址位置。为保留一个额外的字节给存储器寻址，第 17 位 (A16) 被映射到从组件地址选择位 (A0)。专用的两个地址字节表示 A0 至 A15 位。

**图 7. 存储器从组件地址**



### RTC 寄存器从组件

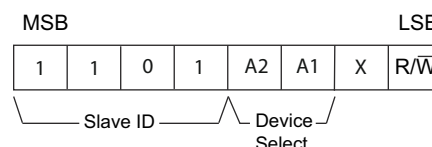
如果主组件发出的从组件地址为 1101b 后跟两位组件选择，则选择 RTC 寄存器进行读 / 写操作。然后，根据从组件地址的 R/W 位，将从 RTC 寄存器读取数据（R/W = “1”）或向 RTC 寄存器写

入数据（R/W = “0”）。RTC 寄存器的从组件地址后跟 RTC 寄存器的一个字节地址以进行读 / 写操作。表 10 中介绍了 RTC 寄存器映射。

组件地址寄存器中的前四个最高有效位 [7:4] 可用于选择 nvSRAM 功能。

入数据（R/W = “0”）。RTC 寄存器的从组件地址后跟 RTC 寄存器的一个字节地址以进行读 / 写操作。表 10 中介绍了 RTC 寄存器映射。

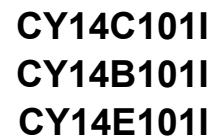
**图 8. RTC 寄存器从组件地址**



### 控制寄存器从组件

控制寄存器的从组件包括序列号、产品 ID、存储器控制和命令寄存器。

如果主组件发出的从组件地址为 0011b 后跟两位组件选择，则选择 nvSRAM 控制寄存器从组件进行读 / 写操作。然后，根据从组件地址的 R/W 位，将从组件读取数据（R/W = “1”）或向组件写入数据（R/W = “0”）。



MSB LSB

0	0	1	1	A2	A1	X	R/W
---	---	---	---	----	----	---	-----

Slave ID
  Device Select

地址	说明	读 / 写	详细信息
0x00	存储器控制寄存器	读 / 写	包含模块保护位和序列号锁定位
0x01	序列号 8 字节	读 / 写（设置 SNL 时为只读）	可编程序列号 通过将存储器控制寄存器中的序列号锁定位设置为“1”进行锁定。
0x02			
0x03			
0x04			
0x05			
0x06			
0x07			
0x08			
0x09	组件 ID	只读	组件 ID 是出厂时设置的
0x0A			
0x0B			
0x0C			
0x0D	保留	保留	保留
0xAA	命令寄存器	只写	允许用于存储、回读、自动存储启用 / 禁用、睡眠模式的命令

存储器控制寄存器包含以下位:

位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0	SNL (0)	0	0	BP1 (0)	BP0 (0)	0	0

- **BP1:BP0**: 模块保护位可用于保护 1/4、1/2 或全存储阵列。可通过写入指令将这些位写入控制寄存器从组件的 0x00 位置。但是，无论模块是否被保护，任何存储循环都会将 SRAM 数据传输到非易失性单元中。BP0 和 BP1 的默认出厂设置值为“0”。

电平	BP1:BP0	模块保护
0	00	None
1/4	01	0x18000~0x1FFFF
1/2	10	0x10000~0x1FFFF
1	11	0x00000~0x1FFFF

- | 数据字节<br>[7:0] | 命令            | 说明                   |
|---------------|---------------|----------------------|
| 0011 1100     | STORE         | 将 SRAM 数据存储到非易失性存储器中 |
| 0110 0000     | RECALL        | 将数据从非易失性存储器回读到 SRAM  |
| 0101 1001     | ASENB         | 启用自动存储               |
| 0001 1001     | ASDISB        | 禁用自动存储               |
| 1011 1001     | SLEEP<br>(睡眠) | 进入睡眠模式以实现低功耗         |

- 注意:** 如果禁用自动存储, 且不需要  $V_{CAP}$ , 则必须让  $V_{CAP}$  引脚保持开路状态。  $V_{CAP}$  引脚绝对不可以接地。在任何情况下都不能禁用加电回读操作。



■ **SLEEP:** 可通过 SLEEP 指令使 nvSRAM 进入睡眠模式。寄存 SLEEP 指令后, nvSRAM 需要花费  $t_{SS}$  的时间来处理 SLEEP 请求。在成功寄存和处理 SLEEP 命令后, nvSRAM 将 HSB 切换为低电平, 执行存储操作将数据保存到非易失性存储器中, 然后进入睡眠模式。只要进入睡眠模式, nvSRAM 就会启动非易失性存储循环, 这样每次执行睡眠命令后都会失去一次耐久性循环。存储循环仅在上一个存储或回读循环之后已经执行对 SRAM 写操作时开始。

nvSRAM 采用以下方式进入睡眠模式:

1. 主组件发送 START 命令。
2. 在设置了 I<sup>2</sup>C 写位 (R/W = "0") 的情况下主组件发送控制寄存器从组件 ID。
3. 从组件 (nvSRAM) 将 ACK 发送回主组件。
4. 主组件发送命令寄存器地址 (0xAA)。
5. 从组件 (nvSRAM) 将 ACK 发送回主组件。
6. 主组件发送命令寄存器字节以进入睡眠模式。
7. 从组件 (nvSRAM) 将 ACK 发送回主组件。
8. 主组件生成 STOP 条件。

进入睡眠模式后, 睡眠指令寄存后, 组件会在寄存 SLEEP 指令后开始消耗  $I_{ZZ}$  电流  $t_{SLEEP}$  长的时间。在组件退出睡眠模式之前, 无法对组件进行正常操作访问。主组件传输从组件地址后, nvSRAM 将在  $t_{WAKE}$  时间后唤醒。

传输三个从组件地址中的任何一个都可将 nvSRAM 从睡眠模式唤醒。在  $t_{SLEEP}$  和  $t_{WAKE}$  时间间隔内, 无法访问 nvSRAM 组件。主组件对 nvSRAM 的任何访问尝试将被忽略, nvSRAM 向主组件发送 NACK。另一种确定组件何时准备好的方法是主组件发送读或写命令并寻找 ACK。

### 写保护 (WP)

写保护 (WP) 引脚为有效高电平引脚, 防止整个存储器和所有寄存器执行写操作。为了禁止所有写操作, 该引脚必须保持高电平。当该引脚处于高电平时, 所有存储器和寄存器写操作被禁止并且地址计数器不递增。该引脚在内部被置于低电平, 因此, 不用时可将其保持开路状态。

### 自动存储操作

自动存储操作是 nvSRAM 的独有特性, 可在断电期间自动将 SRAM 数据存储到 QuantumTrap 单元中。该存储利用外部电容 ( $V_{CAP}$ ), 在断电时, 让组件可以安全地将数据存储到非易失性存储器中。

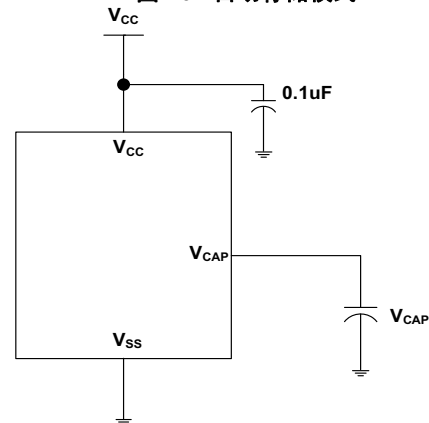
在正常工作时, 组件从  $V_{CC}$  得到电流以对与  $V_{CAP}$  引脚连接的电容充电。在断电期间, 当  $V_{CC}$  引脚上的电压降至  $V_{SWITCH}$  以下时, 组件将禁止对 nvSRAM 的所有存储器访问, 并使用来自

$V_{CAP}$  电容的电荷自动执行有条件的存储操作。如果在上一次存储或回读之后未执行写循环, 则不启动自动存储操作。

**注意:** 如果电容未连接至  $V_{CAP}$  引脚, 则必须通过发出第 8 页上的命令寄存器中指定的自动存储禁用指令禁用自动存储。如果在没有  $V_{CAP}$  引脚上的电容的情况下启用自动存储, 则组件将在没有足够电荷的情况下尝试自动存储操作以完成存储。这将破坏 nvSRAM 中存储的数据和序列号, 并使 SNL 位解锁。

图 10 显示自动存储操作之存储电容 ( $V_{CAP}$ ) 的正确连接。有关  $V_{CAP}$  的大小, 请参考第 29 页上的直流电气特性。

图 10. 自动存储模式



### 硬件存储和 HSB 引脚操作

CY14X101I 中的 HSB 引脚用于控制和确认存储操作。如果没有进行中的存储或回读, 该引脚可以用于请求硬件存储循环。当 HSB 引脚被置为低电平时, 组件在  $t_{DELAY}$  时间之后有条件地启动存储操作。存储循环仅在上一个存储或回读循环之后已经执行对 SRAM 写操作时开始。对存储器的读和写操作将被禁止  $t_{STORE}$  时间或在 HSB 引脚为低电平时都被禁止。

HSB 引脚还起到开漏驱动器 (内部 100 kΩ 弱上拉电阻) 的作用, 它在进行 (通过任何手段触发的) 存储时在内部变为低电平以指示繁忙状态。

**注意:** 在每次硬件和软件存储操作之后, HSB 通过标准输出高电流变为高电平一小段时间 ( $t_{HHHD}$ ), 然后通过内部 100 kΩ 上拉电阻一直保持高电平。

**注意:** 若想成功进行上一个数据字节存储, 硬件存储必须在收到上一个数据位 D0 之后的至少一个时钟周期时启动。

存储操作完成后, 在 HSB 引脚变回高电平后, nvSRAM 存储器访问将被禁止  $t_{LZHSB}$  的时间。如果不使用 HSB 引脚, 让其保持未连接状态即可。

## 硬件回读（加电）

加电期间，当  $V_{CC}$  超过  $V_{SWITCH}$  时，将启动自动回读序列，将非易失性存储器的内容传输到 SRAM。数据之前可能已通过存储序列存储在非易失性存储器中。

加电回读循环需要  $t_{FA}$  的时间才能完成，在此期间，存储器访问将被禁用。HSB 引脚可用于检测组件的就绪状态。

## 写操作

从组件地址的最后一位指示读或写操作。在执行写入操作时，从组件地址后跟存储器或寄存器地址和数据。只要主组件生成 STOP 或重复 START 条件，或 nvSRAM 发出 NACK，则继续执行写操作。

在以下条件下，nvSRAM 发出 NACK：

1. 未收到有效的组件 ID。
2. 对受保护的存储器模块地址进行写（突发写入）访问时，收到数据字节后，将从 nvSRAM 返回 NACK。然而，地址计数器被设为该地址，下一个当前读操作将从该地址开始。
3. 对无效或超出边界的存储器地址进行写 / 随机读访问时，收到后地址，将从 nvSRAM 返回 NACK。在这种情况下，地址计数器保持不变。

nvSRAM 发送 NACK 后，将终止写操作，同时，SDA 线上的任何数据将被忽略，直到主组件生成 STOP 或重复 START 条件。

例如，考虑这样一种情况：对控制寄存器从组件地址 0x01 执行突发写入访问以写入序列号，继续对地址 0x09（只读寄存器）执行突发写入访问。组件将返回 NACK，地址计数器不递增。下一个读操作从地址 0x09 开始。此外，任何从写保护地址（例如，0x09）开始的写操作将在数据字节被成功发后由 nvSRAM 使用 NACK 进行响应，然后将地址计数器设置为该地址。在这种情况下，下一个读操作也从地址 0x09 开始。

**注意：**如果试图对一个不存在的地址（例如，控制寄存器从组件中的 0x0D 或 RTC 寄存器中的 0x3F）进行读 / 写访问，nvSRAM 将在发送超出边界的地址后，使用 NACK 立即进行响应。地址计数器保持不变，仍然保持上一个成功的读或写操作地址。

传输数据第 8 位后，立即内部执行写操作（无任何延迟）。如果写操作不是想要执行的操作，主组件必须在第八个时钟周期前通过生成 STOP 或重复 START 条件终止写操作。

有关写指令的详细信息，请参见[存储器从组件访问](#)一节。

## 读操作

如果从组件地址的最后位是“1”，则采取读操作，nvSRAM 在主组件发送从组件地址字节后立即控制 SDA 线路。读操作从当前地址位置（上一个成功的读或写操作地址之后的地址）开始。到达最后一个地址后，地址计数器重新循环到第一个地址。

如果为控制寄存器从组件，则无论何时执行突发写入以进入不存在的地址，读操作将重新循环到 0x00。这对于命令寄存器尤其适用。

可使用下列方法终止读操作：

1. 主组件在第九个时钟周期发送 NACK，紧接着在第十个时钟周期发送 STOP 或重复 START 条件。
2. 主组件在第九个时钟周期生成 STOP 或重复 START 条件。

有关写指令的详细信息，请参见[存储器从组件访问](#)一节。

## 存储器从组件访问

以下各节介绍从 nvSRAM 执行读或写操作所需的数据传输序列。

### 写入 nvSRAM

每个写操作包括生成开始条件后传输的从组件地址。从组件地址的最后位必须设置为“0”，以指示写操作。主组件可写入一个数据字节，或继续写入多个连续的地址位置，同时内部地址计数器自动递增。访问存储器中的最后一个地址后，地址寄存器重置为 0x00000。继续执行写操作，直到主组件生成 STOP 或重复 START 条件，或 nvSRAM 发出 NACK。

只有当 nvSRAM 收到所有八个数据位后，才执行写操作。成功执行写操作后，nvSRAM 将发送 ACK 信号。主组件生成 STOP 条件或重复 START 操作可终止写操作。如果主组件希望终止当前写操作而不改变存储器内容，应在第八个数据位前使用 START/STOP 条件执行该操作。

如果主组件尝试访问 nvSRAM 上的写保护存储器地址，则传输要写入受保护地址的数据字节后返回 NACK，同时地址计数器不会递增。类似地，执行突发模式的写操作时，发送尝试写入受保护的存储器位置的数据字节后返回 NACK，同时地址计数器不会递增。

图 11. nvSRAM 单字节写入 (Hs 模式除外)

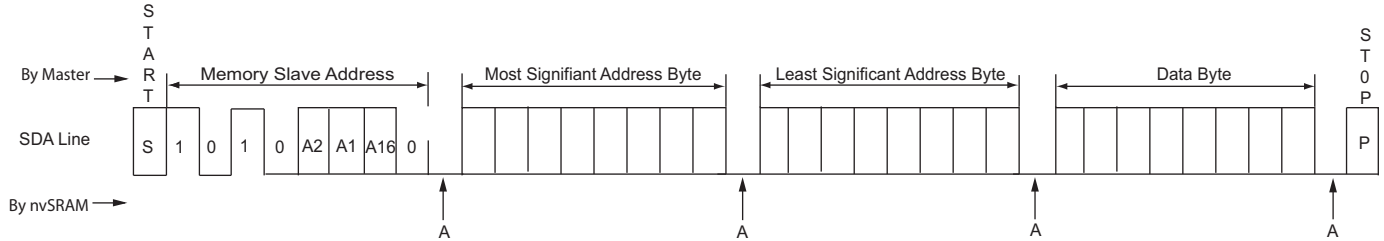


图 12. nvSRAM 多字节写入 (Hs 模式除外)

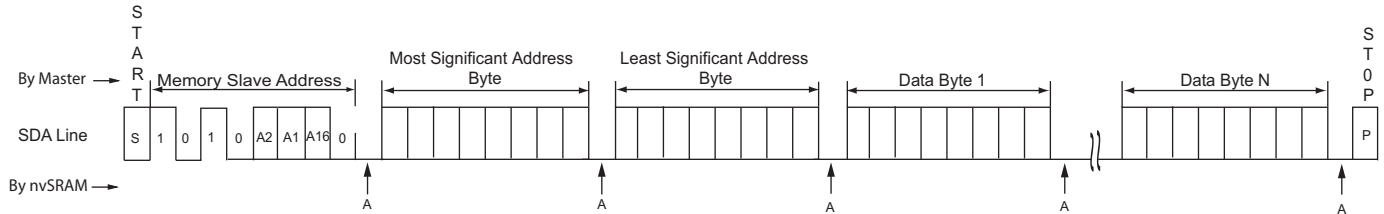


图 13. nvSRAM 单字节写入 (Hs 模式)

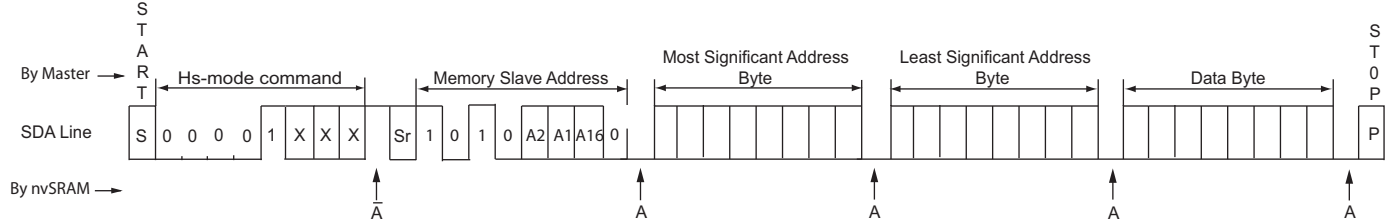
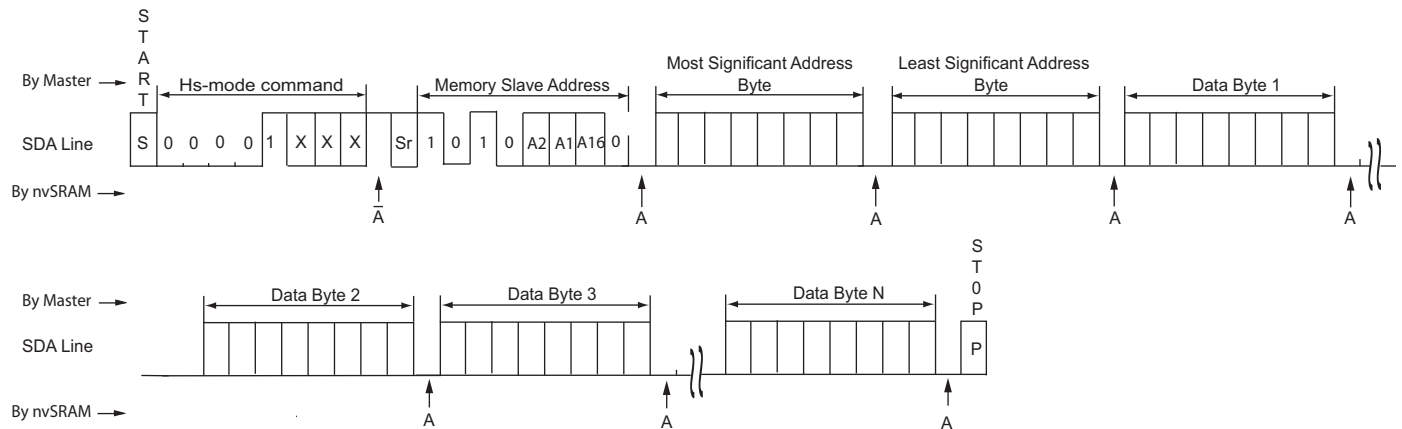


图 14. nvSRAM 多字节写入 (Hs 模式)

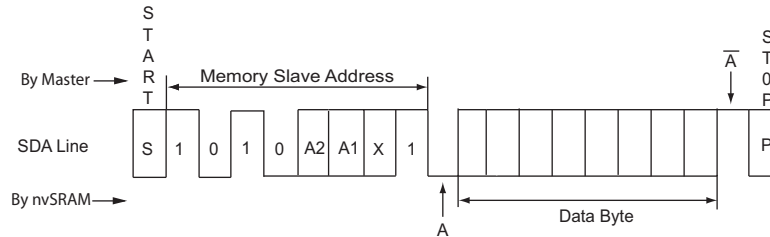


### 当前 nvSRAM 读取

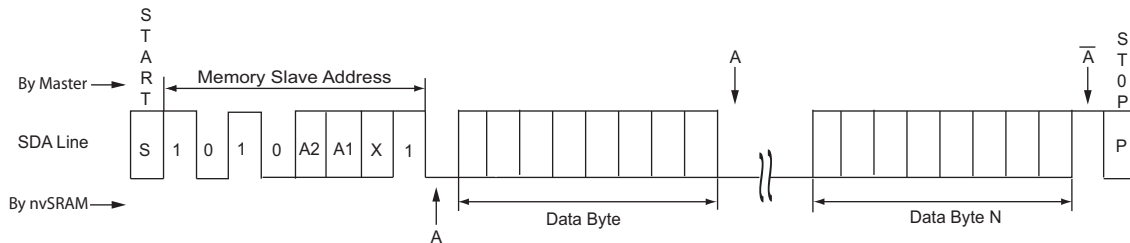
每个读操作都以主组件传输 nvSRAM 从组件地址同时 LSB 设为“1”以指示“读取”开始。读取操作从地址计数器中的地址开始。地址计数器设为最后一次“写”或“读”操作访问的地址的下一个地址位置。主组件读取 1 字节后可终止读操作或继续按顺序读取地址，直到存储器中的最后一个地址，然后地址计数器返回至地址 0x00000。有关终止读访问的有效方法，请参见第 10 页上的读操作一节。

**注意：**进行当前 nvSRAM 读访问时，将忽略 A16 位。

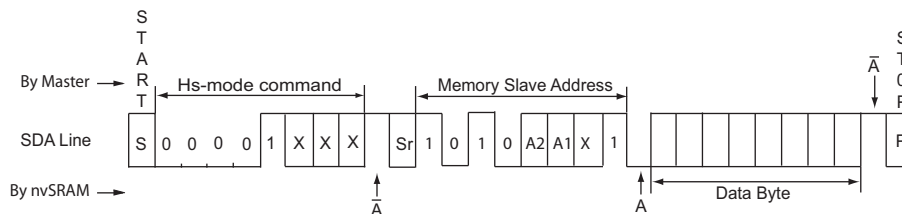
**图 15. 当前位置单字节 nvSRAM 读取（Hs 模式除外）**



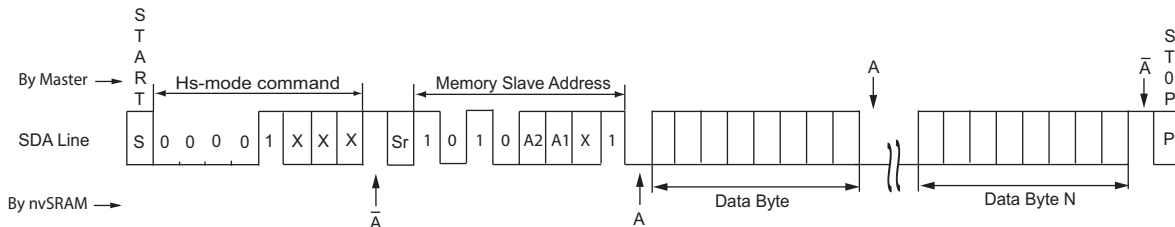
**图 16. 当前位置多字节 nvSRAM 读取（Hs 模式除外）**



**图 17. 当前位置单字节 nvSRAM 读取（Hs 模式）**



**图 18. 当前位置多字节 nvSRAM 读取（Hs 模式）**



### 随机地址读取

通过最后地址字节被确认后立即启动写操作并生成重复 START 条件来执行随机地址读取。地址计数器设置为该地址，对该从组件的下一个读访问从该地址启动读操作。主组件读取 1 字节后可终止读操作或继续按顺序读取地址，直到存储器中的最后一个地址，然后地址计数器返回至开始地址 0x00000。

图 19. 随机地址单字节读取（Hs 模式除外）

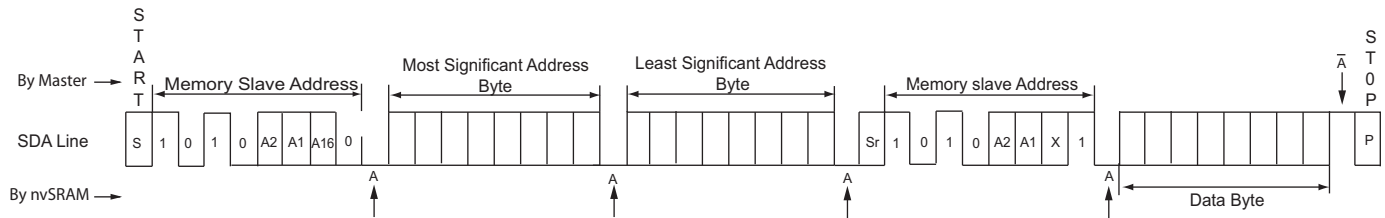


图 20. 随机地址多字节读取（Hs 模式除外）

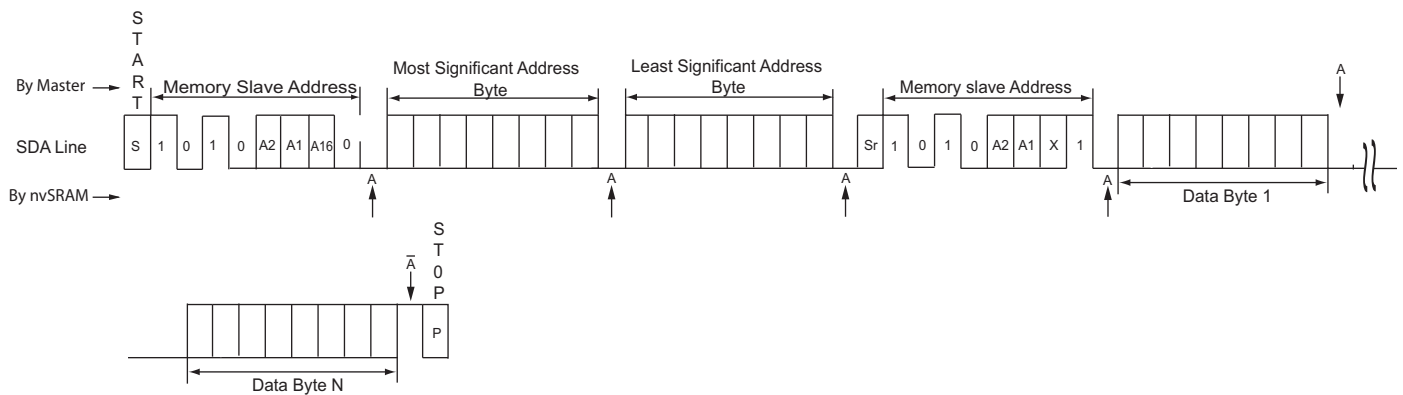


图 21. 随机地址单字节读取（Hs 模式）

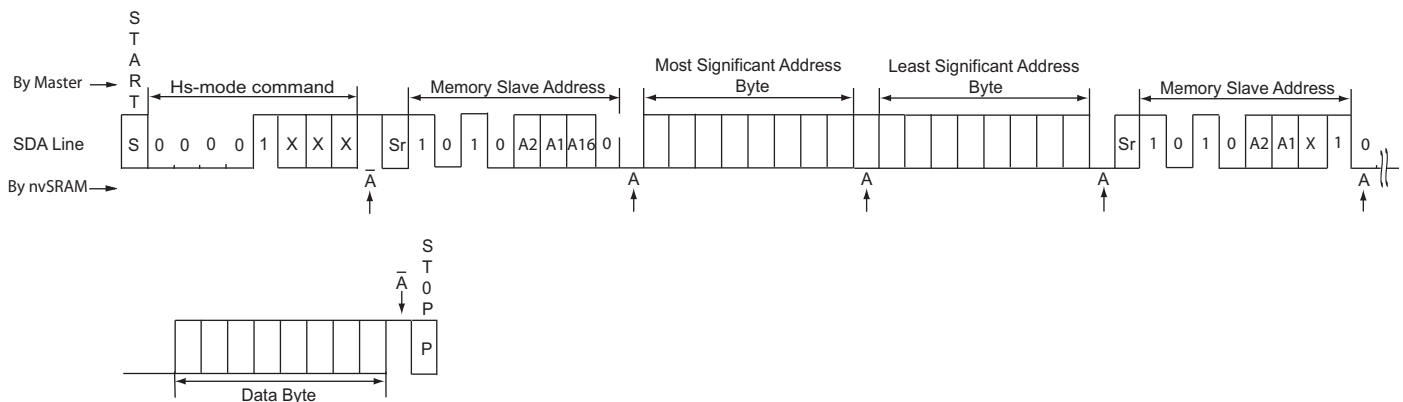
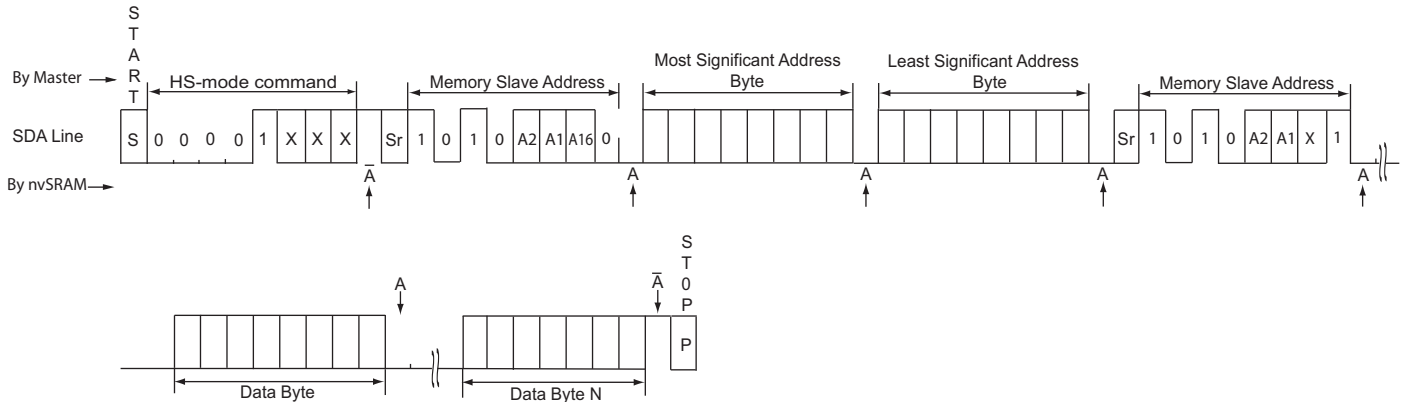


图 22. 随机地址多字节读取 (Hs 模式)



### RTC 寄存器从组件访问

以下各节介绍从 RTC 寄存器执行读或写操作所需的数据传输序列。

#### 写入 RTC 寄存器

使用 RTC 寄存器从组件地址后跟一个字节的地址和数据启动对 RTC 寄存器的写操作。主组件可写入一个数据字节，或继续写入多个连续的地址位置，同时内部地址计数器自动递增。访问最后一个 RTC 寄存器后，地址寄存器重置为 0x00。继续执行写操作，直

到主组件生成 STOP 或重复 START 条件，或 nvSRAM RTC 寄存器从组件发出 NACK。

只有当 nvSRAM 收到所有八个数据位后，才执行写操作。成功执行写指令操作后，nvSRAM 发出 ACK 信号。主组件在传输最后数据位前生成 STOP 条件或重复 START 操作可终止写操作。

如果主组件尝试访问 RTC 寄存器从组件中超出边界的存储器地址，传输地址位后返回 NACK。地址计数器保持不变，下一个当前读操作从地址计数器中保持的地址值开始。

图 23. RTC 寄存器单字节写入

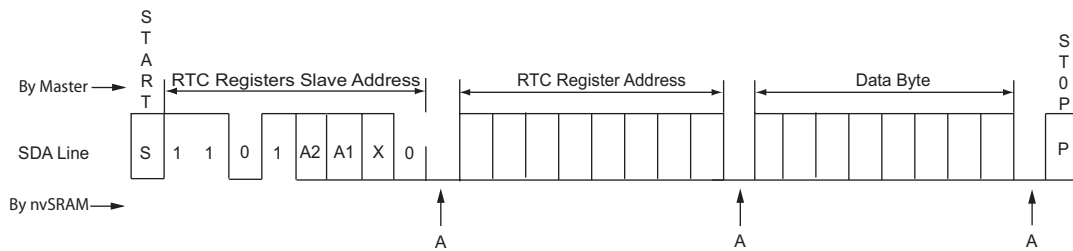
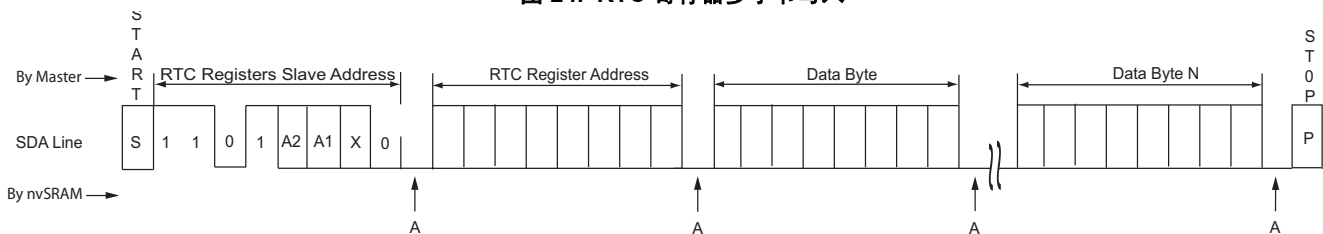


图 24. RTC 寄存器多字节写入





### 当前地址 RTC 寄存器读取

RTC 寄存器当前读取以主组件在生成 START 条件后发送 RTC 寄存器从组件地址开始。所有读操作从当前地址（上一个被访问地址位置的下一个地址）开始。按顺序读取最后一个地址后，地址锁存重新循环至第一个位置 (0x00)，继续进行读操作。主组件读取 1 字节后可终止读操作或继续按顺序读取地址，直到存储器中的最后一个地址，然后地址计数器返回至地址 0x00。主组件生成 STOP 条件或重复 START 操作或发送 NACK 可终止写操作。

图 25. 当前地址 RTC 寄存器单字节读取

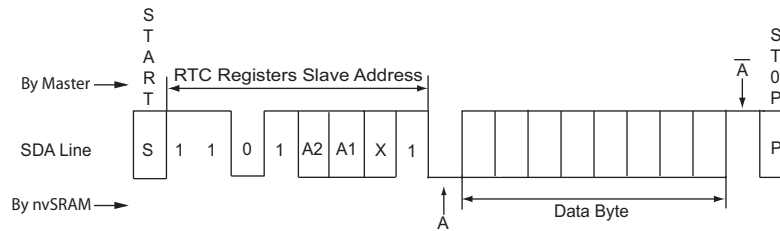
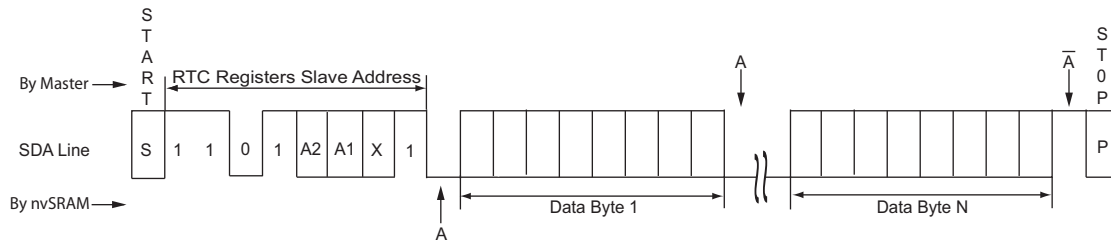


图 26. 当前地址 RTC 寄存器多字节读取



### 随机地址 RTC 寄存器读取

通过最后地址字节被确认后立即启动写操作并生成重复 START 条件来执行随机地址读取。地址计数器设置为该地址，对该从组件的下一个读访问从该地址启动读操作。主组件读取 1 字节后可终止读操作或继续按顺序读取地址，直到存储器中的最后一个地址，然后地址计数器返回至 RTC 的开始地址位置 (0x00)。

传输地址字节后由 nvSRAM 返回 NACK 来响应对 RTC 寄存器从组件中超出边界的存储器地址进行的随机地址读取尝试。地址计数器保持不变，下一个当前读操作从地址计数器中保持的地址值开始。

图 27. 随机地址 RTC 寄存器单字节读取

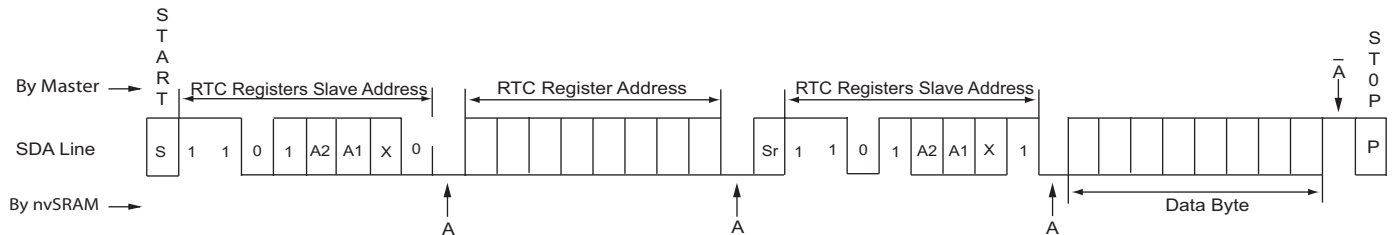
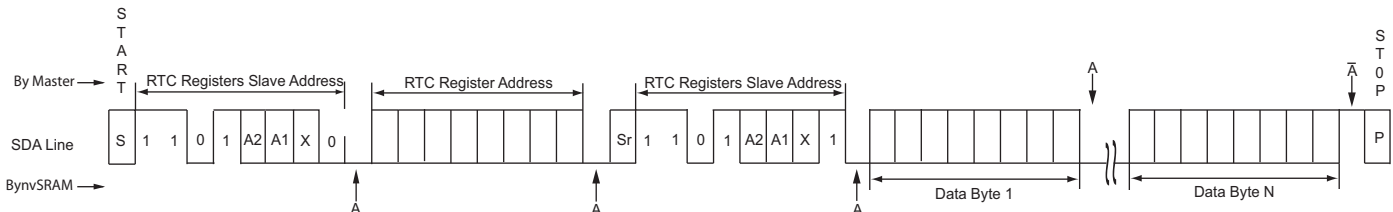


图 28. 随机地址 RTC 寄存器多字节读取



## 控制寄存器从组件

以下各节介绍从控制寄存器从组件执行读或写操作所需的数据传输序列。

### 写入控制寄存器

要对控制寄存器从组件执行写入操作，主组件将在生成 START 条件后传输控制寄存器从组件地址。写入序列从主组件指定的地址位置开始，直到主组件生成 STOP 条件或最后一个可写入地址位置。

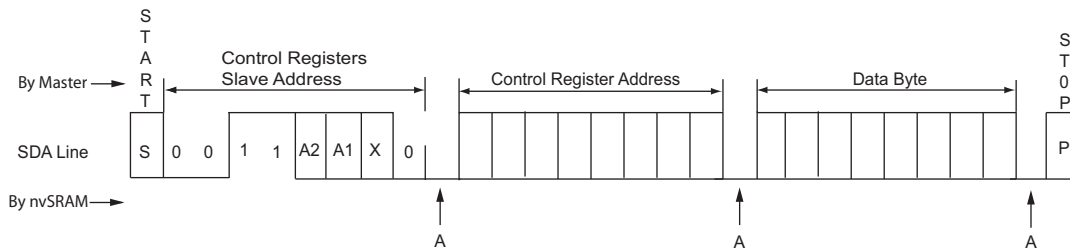
在正常或突发写入过程中，执行写操作时如果访问不可写的地址位置，数据字节被发送后从组件将生成 NACK 并且写序列将终止。后续的所有数据字节都将被忽略，地址计数器不递增。

如果对命令寄存器 (0xAA) 执行写操作，下一个当前读操作也从第一个地址 (0x00) 开始 (与此处的情况一样)。当前地址为超出边界的地址。地址不递增，下一个当前读操作从该地址位置开始。如果对超出边界的地址位置尝试写操作，nvSRAM 将在地址字节被发送后立即发送 NACK。

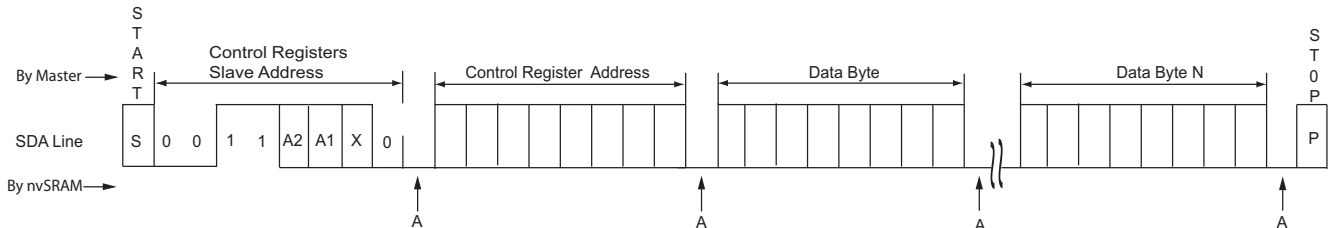
此外，如果序列号被锁定，控制寄存器从组件中只有两个地址 (0xAA 或命令寄存器以及 0x00 或存储器控制寄存器) 可写入。对任何其他地址位置进行写入操作时，组件将确认命令字节和地址字节，但针对数据字节从控制寄存器从组件返回 NACK。在这种情况下，地址不会递增，当前读取从上一个被确认的地址开始。

当主组件对超出边界的存储器地址进行写操作访问时，nvSRAM 控制寄存器从组件发送 NACK。在这种情况下，下一个当前读取操作从上一个被确认的地址开始。

**图 29. 控制寄存器单字节写入**



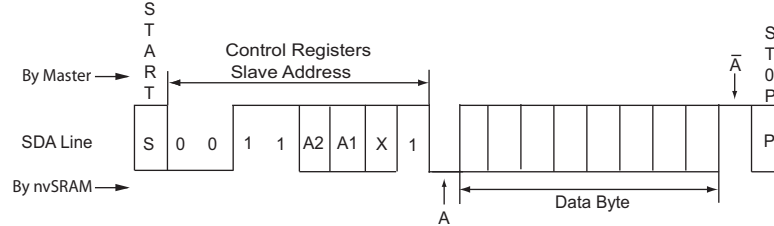
**图 30. 控制寄存器多字节写入**



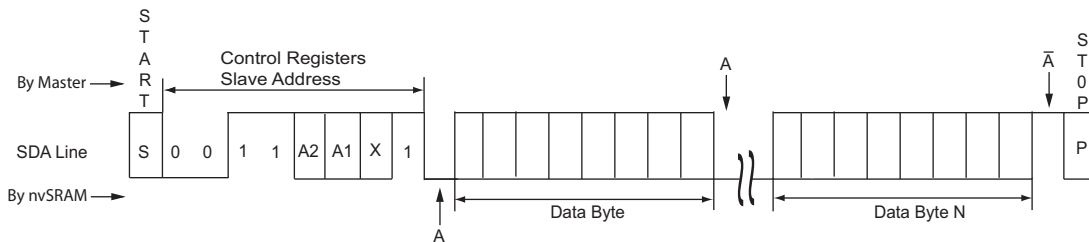
### 当前控制寄存器读取

控制寄存器从组件读操作以主组件在生成 START 条件且 LSB 设置为“1”之后发送控制寄存器从组件地址开始。该读取操作从当前地址（最后访问地址位置后的地址）开始。控制寄存器从组件地址读取继续，直到最后可读地址位置，然后重新循环至第一个位置 (0x00)。注意：命令寄存器为只写寄存器，无法通过顺序读操作进行访问。如果突发读操作从命令寄存器 (0xAA) 开始，地址计数器将折返至寄存器映射中的第一个地址 (0x00)。

**图 31. 控制寄存器单字节读取**



**图 32. 当前控制寄存器多字节读取**



### 随机控制寄存器读取

通过对预期的读位置启动写操作并立即执行重复 START 操作可执行随机地址读取。控制寄存器从组件地址读取继续，直到最后可读地址位置，然后重新循环至第一个位置 (0x00)。注意：命令

寄存器为只写寄存器，无法通过顺序读操作进行访问。从命令寄存器 (0xAA) 开始的随机读取重新循环至控制寄存器寄存器映射中的第一个地址 (0x00)。如果从超出边界的存储器地址开始进行读操作，nvSRAM 将在地址字节被发送后立即发送 NACK。

**图 33. 随机控制寄存器单字节读取**

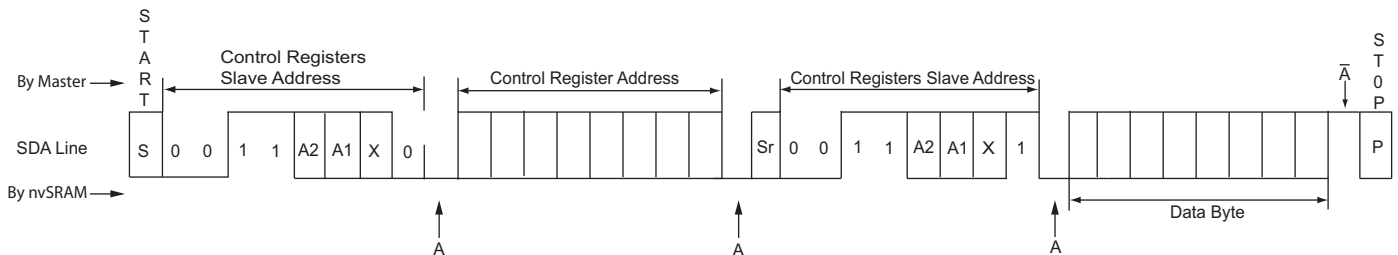
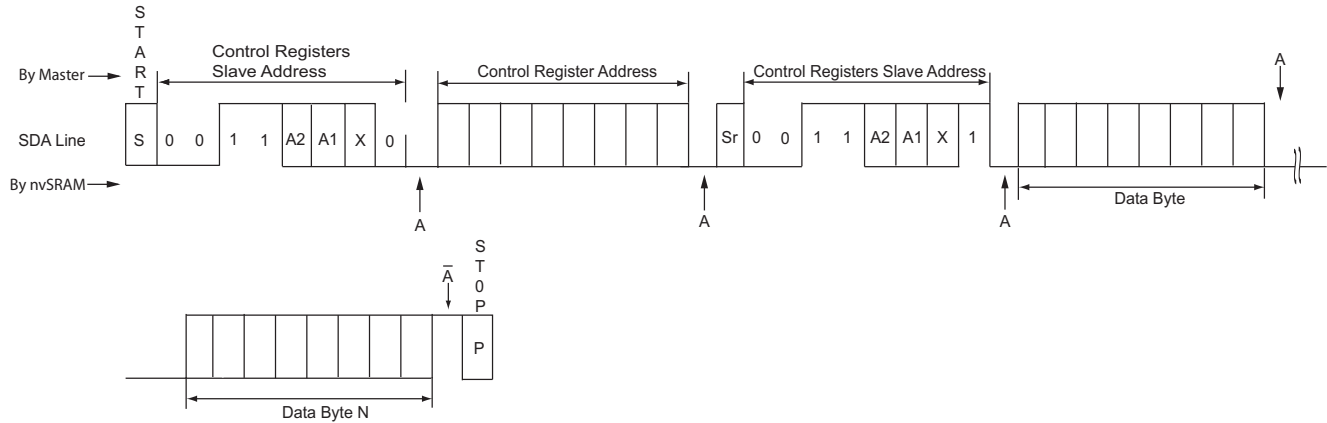


图 34. 随机控制寄存器多字节读取



## 序列号

序列号为提供给用户以对该组件进行唯一标识的 8 字节存储空间。它通常由两个字节的客户 ID、五个字节的唯一序列号及一个字节的 CRC 校验组成。然而，nvSRAM 不计算 CRC 值，由用户来决定以所需的格式使用 8 字节的存储空间。八字节位置的默认值设置为“0x00”。

## 序列号写入

通过控制寄存器从组件可访问序列号。要写入序列号，主组件在生成 START 条件后发送控制寄存器从组件地址，并从 0x01 地址位置开始进行写入，直到 0x08 地址。序列号寄存器内容在下一个存储操作时被保存在非易失性存储器中。如果启用自动存储，断电时 nvSRAM 自动将序列号保存在非易失性存储器中。然而，如果自动存储被禁用，用户必须执行存储操作以保存序列号寄存器的内容。

**注意：**如果未设置序列号锁定 (SNL) 位，则无论是否执行了存储操作，序列号寄存器都可能被重写。一旦设置了序列号锁定位，将禁止对序列号寄存器进行写操作。如果在设置了锁定位时主组件尝试对序列号寄存器执行写操作，则返回 NACK 并且不执行写操作。

## 序列号锁定

对序列号寄存器的写操作完成后，主组件负责通过将存储器控制寄存器 (0x00) 中的序列号锁定位设置为“1”来锁定序列号。存储器控制寄存器中的内容和序列号在下一个存储操作（存储或自动存储）时被存入。如果未启用自动存储，用户必须执行存储操作以保存锁定位状态。

如果未执行存储操作，则序列号锁定位在电源循环后会发生改变。序列号锁定位和 8 字节序列号在加电时默认为“0”。

## 序列号读取

通过对控制寄存器从组件执行读操作可以读回序列号。控制寄存器组件在执行突发读取操作时从最后一个地址（命令寄存器除外）重新循环至 0x00 地址位置。序列号驻留在位置 0x01 至 0x08 中。即使序列号未被锁定，序列号读操作也将返回写入到序列号寄存器中的当前值。设置锁定位前，主组件可以执行序列号读操作以确认写入寄存器的序列号是否正确。

## 组件 ID

组件 ID 是由 JEDEC 分配的制造商 ID、产品 ID、密度 ID 和 die 修订版组成的 4 字节代码。这些寄存器在出厂前已设置好，对用户而言为只读寄存器。

**表 6. 组件 ID**

组件	组件 ID (4 字节)	组件 ID 描述			
		31–21 (11 位)	20–7 (14 位)	6–3 (4 位)	2–0 (3 位)
		制造商 ID	产品 ID	密度 ID	Die 修订版
CY14C101I	0x0681E2A0	00000110100	00001111000101	0100	000
CY14B101I	0x0681EAA0	00000110100	00001111010101	0100	000
CY14E101I	0x0681F2A0	00000110100	00001111100101	0100	000

组件 ID 分为四部分，如表 6 中所示：

### 1. 制造商 ID（11 位）

这是由 JEDEC 为赛普拉斯分配的制造商 ID。JEDEC 将制造商 ID 分成不同的组。制造商 ID 的前三位表示 ID 所属的组。接下来的八位表示制造商 ID。

赛普拉斯制造商 ID 为 0x34，所属的组为 0。因此，所有赛普拉斯 nvSRAM 产品的制造商 ID 如下所示：

赛普拉斯 ID — 000\_0011\_0100

### 2. 产品 ID（14 位）

组件的产品 ID 如表 6 所示。

### 3. 密度 ID（4 位）

4 位密度 ID 用法如表 6 所示，代表产品的密度为 1 Mb。

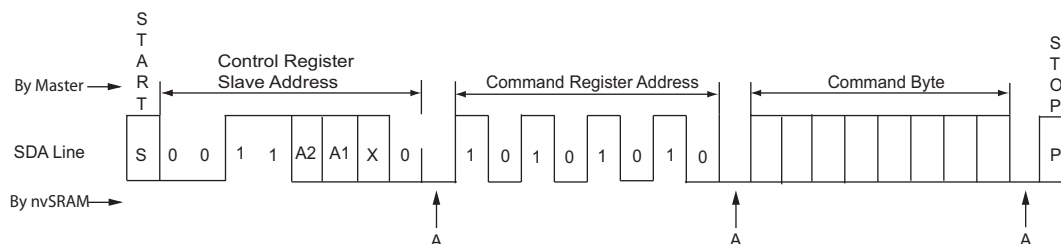
### 4. Die 修订版（3 位）

这用于表示产品设计的任何重大更改。其初始设置始终为 0x0。

## 使用命令寄存器执行命令

控制寄存器从组件允许通过向命令寄存器 (0xAA) 写入特定命令字节来执行不同的命令。每个命令的命令字节代码如表 5 所示。在执行这些命令期间，组件无法访问，如果选择三个从组件中的任何一个，将返回 NACK。如果主组件发送无效命令，nvSRAM 将使用 ACK 进行响应，指示已使用 NOP（无操作）对命令进行了确认。地址翻转至 0x00 位置。

**图 35. 使用命令寄存器执行命令**



## 实时时钟操作

### nvTIME 操作

CY14X101I 提供了具有时钟、警报、看门狗、中断和控制功能的内部寄存器。RTC 寄存器占用单独的 nvSRAM 地址空间，可通过寄存器地址为 0x00 至 0x0F 的 RTC 寄存器读和写序列进行访问。时钟和定时器信息寄存器采用双缓冲技术防止在读或写操作期间访问内部过渡时钟数据。双缓冲技术还避免了在访问时钟数据期间影响正常的定时计数或内部时钟的准确性。时钟和警报寄存器以 BCD 格式存储数据。

### 时钟操作

时钟寄存器以一秒的增量保存时间，最长达 9,999 年。时间可被设置为任何日历时间；时钟自动记录某月某日、某周某日、闰年及世纪转换。专用于时钟功能的寄存器共有八个，这些寄存器用于使用写周期设置时间，以及基于读周期读取时间。这些寄存器包含 BCD 格式的时间。定义为“0”的位目前不可用，这些位被保留以供赛普拉斯将来使用。

### 读取时钟

双缓冲 RTC 寄存器结构降低了从时钟读取错误数据的可能性。当读取位“R”（位于标志寄存器 0x00 位置）设置为“1”时，在读取时钟数据之前对 CY14X101I 计时寄存器的内部更新将停止，以防止读取过渡数据。寄存器更新停止不会影响时钟准确性。

当 RTC 组件读取序列启动后，用户计时寄存器更新将停止，直到生成 STOP 或重复 START 条件后才重新开始更新。内部时钟继续运行的同时读取 RTC 寄存器。读取序列完成后，所有 RTC 寄存器在 20 ms 内同时更新。

### 设置时钟

对 RTC 组件进行写访问将使计时寄存器更新停止，而当写入位“W”（位于标志寄存器 0x00 位置）设置为“1”时允许用户设置时间。然后，正确的星期、日期和时间被写入寄存器（必须为 24 小时 BCD 格式）。写入的时间称为“基准时间”。该值保存在非易失性寄存器中，用于计算当前时间。当通过写入“0”清除写入位并且主组件生成 STOP 或重复 START 条件时，计时寄存器的值被传输到实际的时钟计数器，然后该时钟恢复正常运行。如果主组件未生成有效的 STOP 或重复 START 条件，写入 RTC 寄存器的时间不会传输到实际的时钟计数器。

如果写入 RTC 寄存器的时间不是正确的 BCD 格式，则 RTC 寄存器中每个无效的半字节在翻滚至 0x0 前继续计数至 0xF，然后 RTC 寄存器恢复正常运行。

**注意：**“W”位设置为“0”后，写入计时、警报、校准和中断寄存器的值在  $t_{\text{RTCp}}$  时间后被传输到 RTC 计时计数器中。这些计数器

值必须通过启动软件 / 硬件存储或自动存储操作保存在非易失性存储器中。在“自动存储禁用”模式下， $t_{\text{RTCp}}$  时间后执行存储操作，同时写入 RTC 寄存器以正确记录所做的修改。

### 备用电源

CY14X101I 中的 RTC 用于永久带电操作。在实际应用中，根据是选择了电容还是电池来连接  $V_{\text{RTCcap}}$  或  $V_{\text{RTCbat}}$  引脚。当主电源  $V_{\text{CC}}$  断电并下降至  $V_{\text{SWITCH}}$  以下时，组件切换至备用电源。

时钟振荡器消耗的电流非常少，因此最大程度地延长了后备电源的供电时间。主电源移除后，无论时钟操作如何，存储在 nvSRAM 中的数据都是安全的，因为断电后这些数据被存储在非易失性元件中。

在备用电源操作期间，室温下 CY14X101I 消耗 0.45  $\mu\text{A}$ （典型）的电流。用户必须根据实际应用选择电容或电池值。

下表显示了基于最大电流规格的备用时间。额定备用时间大约比这些时间长两倍。

**表 7. RTC 备用时间**

电容值	备用时间 (CY14B101I)
0.1F	60 小时
0.47F	12 天
1.0F	25 天

使用电容具有明显的优势，即每次系统加电时可对备用电源充电。当主电源移除时，如需备用电池，推荐使用 3V 锂电池和 CY14X101I 电源电流。然而，CY14X101I 在任何时候不会对电池进行充电。必须根据系统生命周期期间总的预期累计断电时间选择电池容量。

### 停止和启动振荡器

校准寄存器中 0x08 位置的 OSCEN 位控制振荡器的启用和禁用。该位是非易失性的，交付给客户时处于“启用”（设置为“0”）状态。系统存放不用时，为了保持电池寿命，OSCEN 必须设置为“1”。这样将关闭振荡器电路，延长电池寿命。如果 OSCEN 位从“禁用”变为“启用”，振荡器大约需要一秒钟（最多两秒）的时间进行启动。

当系统电源关闭时，如果备用电源（ $V_{\text{RTCcap}}$  或  $V_{\text{RTCbat}}$ ）的电压降至各自最低水平以下，振荡器可能掉电。当系统电源恢复时，CY14X101I 能够检测振荡器是否掉电。这记录在标志寄存器 0x00 位置的振荡器掉电标志 (OSCF) 中。当系统通电时（ $V_{\text{CC}}$  大于  $V_{\text{SWITCH}}$ ），会检查 OSCEN 位是否处于“启用”状态。如果 OSCEN 位处于“启用”状态并且振荡器在 5 ms 内未激活，OSCF 位将被设置为“1”。系统必须检查该条件，然后写入“0”以



注意：除设置 OSCF 标志位之外，时间寄存器重置为“基准时间”，这是上次写入计时寄存器的值。控制寄存器或校准寄存器与 OSCEN 位不受“振荡器失效”条件影响。

首次对时间寄存器进行写入时，OSCF 值必须重置为“0”。这将初始化该位的状态（系统首次加电时可能已设置）。

要重置 OSCF，需要将写位“W”（在位于 0x00 的标志寄存器中）设置为“1”以便启用对标志寄存器的写入。对 OSCF 位写入“0”，然后将写位重置为“0”以禁用写操作。

## 校准时钟

RTC 由一个石英控制的晶振以 32.768 kHz 的额定频率驱动。时钟精度取决于晶振和校准的质量。市场中的晶振通常有  $\pm 20$  ppm 到  $\pm 35$  ppm 的误差。然而，CY14X101I 应用一种在 25°C 可将精度提高至  $\pm 1/-2$  ppm 的校准电路。这表示每月有 +2.5 秒到 -5 秒的误差。

校准电路对振荡器分频器电路增加或减少计数来实现此精度。抑制（减少，负校准）或拆分（增加，正校准）的脉冲数量取决于加载到位于 0x08 的校准寄存器中的五个校准位的值。校准位占用校准寄存器中的五个低位。这些位被设置为以二进制形式表示“0”和 31 之间的任何值。D5 位是一个符号位，其中“1”表示正校准，“0”表示负校准。增加计数可使时钟加速，减少计数可使时钟减速。如果将一个二进制“1”加载到寄存器中，其对应振荡器误差中 4.068 或 -2.034 ppm 偏移的调整，具体取决于信号。

校准在 64 分钟周期内发生。对于周期内的头 62 分钟（每分钟一次），可能每 128 个振荡器周期缩短一秒或每 256 个振荡器周期延长一秒。如果将二进制“1”加载到寄存器中，则仅修改 64 分钟周期内的头两分钟。如果将二进制 6 加载到寄存器中，则影响头 12 分钟，以此类推。因此，每个校准步骤对每 125、829 和 120 个实际振荡器周期具有增加 512 个或减少 256 个振荡器周期的作用，即校准寄存器中的每个校准步骤有 4.068 或 -2.034 ppm 的调整。

为了确定所需的校准，标志寄存器 (0x00) 中的 CAL 位必须设置为“1”。这导致 INT 引脚以 512 Hz 的额定频率切换。任何偏离 512 Hz 的偏差表示所需纠正的大小和方向。例如，读数 512.01024 Hz 表示误差为 +20 ppm。因此，必须将十进制值 -10 (001010b) 加载到校准寄存器中以抵消误差。

**注意：**设置或改变校准寄存器不影响测试输出频率。

要将设置或清除 CAL，需要将写位“W”（在位于 0x00 的寄存器中）设置为“1”以便启用对标志寄存器的写入。将某个值写入 CAL，然后将写位重置为“0”以禁用写入。

## 警报

警报功能将用户编写的警报时间值和日期（存储在寄存器 0x01-5 中）与相应的时间和日期值相比较。当出现匹配时，将设置警报内部标志 (AF)，而且如果设置了警报中断启用 (AIE) 位，将在 INT 引脚上生成中断。

有四个警报匹配字段—日期、小时、分和秒。上述每个字段有一个匹配位，用于确定字段是否被用于警报匹配逻辑。将匹配位设置为“0”表示相应的字段用于匹配处理。根据匹配位的不同，警报可以明确到每月发生一次或频繁到每分钟发生一次。不选择任何匹配位（都为 1）表示不需要匹配，因此禁用警报。选择所有的匹配位（都为 0）会导致精确的时间和日期匹配。

可通过两种方法来检测警报事件：读取 AF 标志或监控 INT 引脚。位于标志寄存器 0x00 中的 AF 标志表示发生了日期或时间匹配。当发生匹配时，AF 位设置为“1”。读取标志寄存器会清除警报标志位（和所有其他位）。硬件中断引脚也可能被用于检测警报事件。

要设置、清除或启用警报，需要将“W”位（在标志寄存器 0x00 中）设置为“1”以便启用对警报寄存器的写入。写入警报值后，将“W”位清除为“0”以使更改生效。

**注意：**CY14X101I 要求将针对秒的警报匹配位（警报秒寄存器 0x02 中的“D7”位）设置为“0”以便正确操作警报标志和中断。

## 看门狗定时器

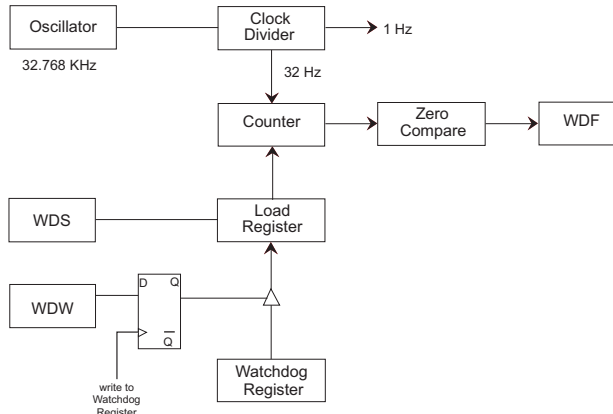
看门狗定时器是一个自由运行且使用从晶体振荡器获得的 32 Hz 时钟 (31.25 ms) 的递减计数器。必须运行振荡器才能使看门狗正常运行。看门狗定时器从看门狗定时器寄存器中加载的值开始递减计数。

定时器由一个可加载的寄存器和一个自由运行的计数器组成。在加电时，寄存器 0x07 中的看门狗超时值加载到计数器加载寄存器中。计数从加电开始并在看门狗探针 (WDS) 位设置为“1”的任何时候从可加载值重新开始。计数器与终止值“0”进行比较。如果计数器达到此值，则产生内部标志和可选中断输出。可以通过在计数器到达“0”之前将 WDS 位设置为“1”来阻止超时中断。这导致计数器重新加载看门狗超时值并重启。只要用户在计数器到达终值之前对 WDS 位进行设置，中断和 WDT 标志就不会产生。

新的时间输出值通过将看门狗写位设置为“0”来写入。当 WDW 值为“0”时，看门狗超时值位 D5-D0 写功能启用以修改超时值。当 WDW 为“1”时，对 D5-D0 位的写入被忽略。WDW 功能使用户能在不考虑看门狗定时器值被修改的情况下设置 WDS 位。看门狗定时器逻辑图如第 22 页上的图 36 所示。注意：将看门狗超时值设置为“0”会禁用看门狗功能。

看门狗定时器的输出为标志位 WDF（如果看门狗允许超时，将设置该标志位）。如果中断寄存器中的看门狗中断启用（WIE）位被设置，看门狗超时将产生 INT 引脚硬件中断。当用户读取标志寄存器时，标志和硬件中断都会被清除。

**图 36. 看门狗定时器框图**



### 可编程方波生成器

方波生成器模块使用晶振输出在组件的 INT 引脚上生成所需的频率。输出频率可编程为以下频率之一：

1. 1 Hz
2. 512 Hz
3. 4096 Hz
4. 32768 Hz

当组件使用备用电源运行时，不生成方波输出。

### 功耗监控器

CY14X101I 提供具有断电中断功能的电源管理方案。它也控制内部开关为时钟提供备用电源并保护存储器在低  $V_{CC}$  条件下不被访问。功耗监控器基于内部带隙参考电路，此电路将  $V_{CC}$  电压与  $V_{SWITCH}$  阈值相比较。

达到  $V_{SWITCH}$  值后，由于  $V_{CC}$  由于断电而发生衰减，将启动从 SRAM 到非易失性元素的数据存储操作，以保存最后的 SRAM 数据状态。电源也从  $V_{CC}$  切换到备用电源（电池或电容）来运行 RTC 振荡器。

当使用备用电源运行时，对 nvSRAM 的读取和写入操作被禁止且 RTC 功能对用户不可用。RTC 时钟继续在后台运行。 $V_{CC}$  存

储到组件后，更新后的 RTC 计时寄存器对用户可用（请参见第 33 页上的 nvSRAM 规范）。

### 备用电源监控器

CY14X101I 提供一个可检测备用电源（备用电池或电容）故障的功耗监控系统。如果发生备用电源故障，备用电源故障标志（BPF）在下次加电时发出。如果备用电压低于  $V_{BAKFAL}$ ，将设置 BPF 标志。即使 RTC 在备用模式下运行，仍会监控备用电源。备用模式运行中检测到的低电压通过 BPF 标志进行标记。BPF 只能将数据保持到出现定义的备用电压 ( $V_{DR}$ ) 低电压电平。

### 中断

CY14X101I 有一个标志寄存器、中断寄存器和将中断信号传送到微控制器的中断逻辑。有三个潜在中断源：看门狗定时器、功耗监控器和警报定时器。通过在中断寄存器（0x06）中适当的设置，可以单独启用上述三个中断源来驱动 INT 引脚。此外，在标志寄存器（0x00）中，每种断源有相应的标志位，主机处理器使用这些标志位来确定中断原因。当中断发生时，INT 引脚驱动器有两个能指定其行为的位。

只有当三个断源中的一个产生中断标志并且各自位于中断寄存器中的中断启用位被启用（设置为“1”）时，才会产生中断。中断源处于活动状态后，两个可编程位 H/L 和 P/L 决定 INT 引脚上输出引脚驱动器的行为。这两个位位于中断寄存器中，可用于驱动 INT 引脚中电平或脉冲模式输出。在脉冲模式中，脉冲宽度内部固定在大约 200 ms。此模式旨在重置主机微控制器。在电平模式中，引脚进入其活动性极，直到用户读取标志寄存器。此模式用作对主机微控制器的中断。下一节对控制位进行了总结。

只有使用常规电源时才生成中断，系统以备用电源模式运行时不会触发中断。

**注意：**只有在加电回读序列完成后，CY14X101I 才生成有效中断。加电后，INT 引脚上的所有事件必须被忽略  $t_{FA}$  的时间。

### 中断寄存器

**看门狗中断启用 (WIE)：**当设置为“1”时，如果发生看门狗超时，看门狗定时器驱动 INT 引脚和一个内部标志。当 WIE 设置为“0”时，看门狗定时器只影响标志寄存器中的 WDF 标志。

**警报中断启用 (AIE)：**当设置为“1”时，警报匹配驱动 INT 引脚和一个内部标志。当 AIE 设置为“0”时，警报匹配只影响标志寄存器中的 AF 标志。

**断电中断启用 (PFE)：**当设置为“1”时，断电监控器驱动引脚和一个内部标志。当 PFE 设置为“0”时，断电监控器只影响标志寄存器中的 PF 标志。

**方波启用 (SQWE)：**当设置为“1”，INT 引脚生成具有可编程频率的方波。频率由中断寄存器的 SQ1 和 SQ0 位决定。该位是非易失性的，在电源循环后保持不变。SQWE 位覆盖所有其他中断。

**高电平 / 低电平 (H/L):** 当设置为“1”时，INT 引脚为高电平有效且驱动器模式为推挽式。只有当  $V_{CC}$  高于  $V_{SWITCH}$  时，INT 引脚才驱动高电平。当设置为“0”时，INT 引脚为低电平有效且驱动器模式为开漏式。INT 引脚必须通过一个 10 k 电阻上拉至  $V_{CC}$ ，同时使用低电平有效模式的中断。

**脉冲 / 电平 (P/L):** 当设置为“1”且发生中断时，会将 INT 引脚驱动约 200 ms。当 P/L 设置为“0”时，INT 引脚驱动至高电平或低电平（由 H/L 决定），直到标志寄存器被读取。

**SQ1 和 SQ0。**当 SQWE 位设置为“1”时，这些位一起用于修正 INT 引脚输出的中波频率。这些位是非易失性的，在电源循环后保持不变。输出频率的决定方式如下表所示。

**表 8. SQW 输出选择**

SQ1	SQ0	频率	注释
0	0	1 Hz	1 Hz 信号
0	1	512 Hz	512 Hz 时钟输出
1	0	4096 Hz	4 kHz 时钟输出
1	1	32768 Hz	振荡器输出频率

当启用的中断源激活 INT 引脚时，外部主机读取标志寄存器来确定原因。请记住，当寄存器被读取时，所有标志都被清除。如果 INT 引脚编程为电平模式，条件将清除并且 INT 引脚返回到其非

活动状态。如果引脚编程为脉冲模式，读取标志也能清除引脚中的标志。如果标志寄存器被读取，脉冲模式不能完成指定的持续时间。如果 INT 引脚用作主机复位，则复位期间标志寄存器不被读取。

以下是显示 INT 引脚状态的概要表。

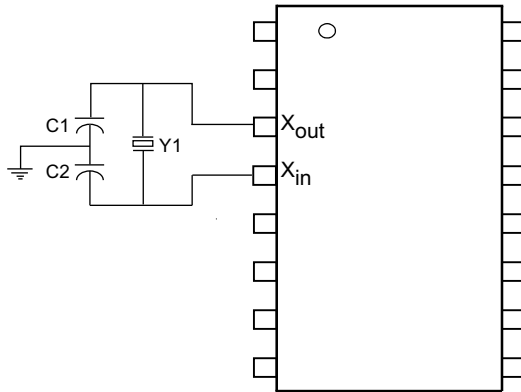
**表 9. INT 引脚状态**

CAL	SQWE	WIE/AIE/PFE	INT 引脚输出
1	X	X	512 Hz
0	1	X	方波输出
0	0	1	警报
0	0	0	HI-Z

### 标志寄存器

标志寄存器有三个标志位：用于生成中断的 WDF、AF 和 PF。这些标志分别由看门狗超时、警报匹配或断电监控器设置。处理器可以轮询此寄存器或使系统通知中断（如果设置了标志）。寄存器被读取后，这些标志会自动复位。在加电时，标志寄存器自动加载值 0x00（OSCF 位除外。请参见第 20 页上的[停止和启动振荡器](#)）。

图 37. RTC 建议的组件配置<sup>[4]</sup>



**Recommended Values**

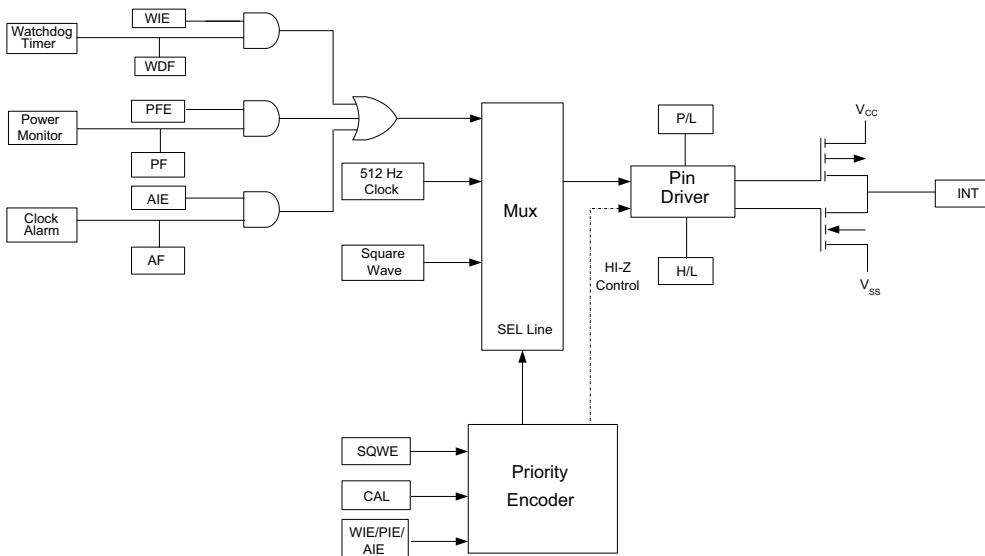
Y1 = 32.768 KHz (12.5 pF)

C<sub>1</sub> = 12 pF

C<sub>2</sub> = 69 pF

**Note:** The recommended values for C1 and C2 include board trace capacitance.

图 38. 中断框图



WDF 一看门狗定时器标志  
WIE 一看门狗中断  
使能  
PF 一断电标志  
PFE 一断电启用  
AF 一警报标志  
AIE 一警报中断启用  
P/L 一脉冲电平  
H/L 一高电平 / 低电平  
SQWE 一方波启用

**注**

4. 有关 nvSRAM RTC 的设计指南和最佳实践，请参见应用笔记 [AN61546](#)。

表 10. RTC 寄存器映射<sup>[5、6]</sup>

寄存器	BCD 格式数据								功能 / 范围
	D7	D6	D5	D4	D3	D2	D1	D0	
0x0F	年 (x10)				年				年：00–99
0x0E	0	0	0	月 (x10)	月				月：01–12
0x0D	0	0	日期 (x10)		日期				日期：01–31
0x0C	0	0	0	0	0	星期			星期：01–07
0x0B	0	0	小时 (x10)		小时				小时：00–23
0x0A	0	分钟 (x10)		分钟				分钟：00–59	
0x09	0	秒 (x10)		秒				秒：00–59	
0x08	OSCEN (0)	0	Cal Sign (0)	校准 (00000)					校准值 <sup>[7]</sup>
0x07	WDS (0)	WDW (0)	WDT (000000)						看门狗 <sup>[7]</sup>
0x06	WIE (0)	AIE (0)	PFE (0)	SQWE (0)	H/L (1)	P/L (0)	SQ1 (0)	SQ0 (0)	中断 <sup>[7]</sup>
0x05	M (1)	0	警报日期 (x10)		警报日期				警报，日期：01–31
0x04	M (1)	0	警报小时 (x10)		警报小时				警报，小时：00–23
0x03	M (1)	警报分钟 (x10)			警报分钟				警报，分钟：00–59
0x02	M (1)	警报秒 (x10)			警报秒				警报，秒：00–59
0x01	世纪 (x10)				世纪				世纪：00–99
0x00	WDF	AF	PF	OSCF <sup>[8]</sup>	BPF <sup>[8]</sup>	CAL (0)	W (0)	R (0)	标志 <sup>[7]</sup>

注

5. ( ) 指明出厂数值。
6. RTC 寄存器未使用的位被保留以供将来使用, 应设置为“0”。
7. 该值为二进制, 而非 BCD 格式。
8. 用户复位 OSCF 和 BPF 标志位后, 应在  $t_{RTCp}$  时间后更新标志寄存器。

表 11. 寄存器映射详细信息

寄存器	说明							
0x0F	计时一年							
	D7	D6	D5	D4	D3	D2	D1	D0
	年 (x10)				年			
	包含年的两个低位 BCD 数字。低位半字节（四位）包含年值；高位半字节（四位）包含以 10 年为单位的值。每半字节的范围为 0 到 9。寄存器的范围是 0 到 99。							
0x0E	计时一月							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	0	0	月 (x10)	月			
	包含月的 BCD 数字。低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（一位）包含高位数字，范围为 0 到 1。寄存器的范围是 1 到 12。							
0x0D	计时一日期							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	0	日期 (x10)		日期			
	包含日期的 BCD 数字。低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（两位）包含 x10 数字，范围为 0 到 3。寄存器的范围是 1 到 31。可针对闰年进行自动调整。							
0x0C	计时一星期							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	0	0	0	0	星期		
	低位半字节（三位）包含一个与星期相关的值。星期是一个环形计数器，从 1 计数到 7，然后返回到 1。用户必须给星期值指定含义，因为星期不与日期集成。							
0x0B	计时一小时							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	0	小时 (x10)		小时			
	包含小时（24 小时格式）的 BCD 值。低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（两位）包含高位数字，范围为 0 到 2。寄存器的范围是 0 到 23。							
0x0A	计时一分钟							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	分钟 (x10)			分钟			
	包含分钟的 BCD 值 低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（三位）包含高位数字，范围为 0 到 5。寄存器的范围是 0 到 59。							
0x09	计时一秒							
	D7	D6	D5	D4	D3	D2	D1	D0
	0	秒 (x10)			秒			
	包含秒的 BCD 值 低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（三位）包含高位数字，范围为 0 到 5。寄存器的范围是 0 到 59。							
0x08	校准 / 控制							
	D7	D6	D5	D4	D3	D2	D1	D0
	OSCEN	0	校准符号	校准				
OSCEN	振荡器启用。当设置为“1”时，振荡器停止。当设置为“0”时，振荡器运行。禁用振荡器可在存储过程中节省电池或电容电源。							
校准符号	确定校准调整是作为对时基增加 (1) 还是作为从时基中减少进行应用 (0)。							
校准	这五个位控制时钟的校准							



表 11. 寄存器映射详细信息 (续)

寄存器	说明							
0x07	看门狗定时器							
	D7	D6	D5	D4	D3	D2	D1	D0
	WDS	WDW	WDT					
WDS	看门狗探针。将该位设置为“1”可重新加载并重启看门狗定时器。将该位设置为“0”不起作用。看门狗定时器复位后，该位被自动清除。WDS 位是只写的。读取该位始终返回 0。							
WDW	看门狗写启用。将此位设置为“1”可禁用对看门狗超时值 (D5–D0) 的任何写入。这使用户可以在不影响超时值的情况下设置看门狗探针位。当下一个写循环完成时，将此位设置为“0”允许 D5–D0 位写入看门狗寄存器。 <a href="#">第 21 页上的看门狗定时器</a> 中详细介绍了该功能。							
WDT	看门狗超时选择。看门狗定时器间隔由该寄存器中的 6 位值选择。它代表一个 32 Hz 计数 (31.25 ms) 的乘数。超时值范围为 31.25 ms（设置为 1）到 2 秒（设置为 3 Fh）。将看门狗定时器寄存器设置为“0”将禁用定时器。仅当在上一个周期中 WDW 位设置为 0 时才能对这些位进行写入。							
0x06	中断状态 / 控制							
	D7	D6	D5	D4	D3	D2	D1	D0
	WIE	AIE	PFE	SQWE	H/L	P/L	SQ1	SQ0
WIE	看门狗中断启用。当设置为“1”时，当看门狗超时发生时，看门狗定时器驱动 INT 引脚和 WDF 标志。当设置为“0”时，看门狗定时器只影响 WDF 标志。							
AIE	警报中断启用。当设置为“1”时，警报匹配驱动 INT 引脚和 AF 标志。当设置为“0”时，警报匹配只影响标志 AF 标志。							
PFE	断电启用。当设置为“1”时，警报匹配驱动 INT 引脚和 AF 标志。当设置为“0”时，断电监视器只影响 PF 标志。							
SQWE	方波启用。当设置为“1”时，在 INT 引脚上以使用 SQ1 和 SQ0 位编程的频率驱动方波。方波输出优先于中断逻辑。如果将 SQWE 设置为“1”，当启用的中断源处于活动状态时，只产生相应的标志，INT 引脚继续驱动方波。							
H/L	高电平 / 低电平。当设置为“1”时，将 INT 引脚驱动为高电平有效。当设置为“0”时，INT 引脚为开漏，低电平有效。							
P/L	脉冲 / 电平。当设置为“1”时，INT 引脚由一个约 200 ms 的中断源驱动有效（由 H/L 决定）。当设置为“0”时，INT 引脚驱动到有效电平状态（由 H/L 设置），直到标志寄存器被读取。							
SQ1, SQ0	SQ1, SQ0。当 SQWE 设置为“1”时，这些位用于决定 INT 引脚输出上的方波频率。以下为每个 (SQ1, SQ0) 组合的频率输出： (0, 0) — 1 Hz (0, 1) — 512 Hz (1, 0) — 4096 Hz (1, 1) — 32768 Hz							
0x05	警报—日期							
	D7	D6	D5	D4	D3	D2	D1	D0
	M	0	警报日期 (x10)			警报日期		
	包含警报日期值和用于选择或取消选择日期值的掩码位。							
M	匹配。当此位设置为“0”时，在警报匹配中使用日期值。将此位设置为“1”会导致匹配电路忽略日期值。							
0x04	警报—小时							
	D7	D6	D5	D4	D3	D2	D1	D0
	M	0	警报小时 (x10)			警报小时		
	包含警报小时值和用于选择或取消选择小时值的掩码位。							
M	匹配。当此位设置为“0”时，在警报匹配中使用小时值。将此位设置为“1”会导致匹配电路忽略小时值。							
0x03	警报—分钟							
	D7	D6	D5	D4	D3	D2	D1	D0
	M	警报分钟 (x10)			警报分钟			
	包含警报分钟值和用于选择或取消选择分钟值的掩码位。							
M	匹配。当此位设置为“0”时，在警报匹配中使用分钟值。将此位设置为“1”会导致匹配电路忽略分钟值。							

表 11. 寄存器映射详细信息 ( 续 )

寄存器	说明							
0x02	警报一秒							
	D7	D6	D5	D4	D3	D2	D1	D0
	M	警报秒 (x10)				警报秒		
	包含警报秒值和用于选择或取消选择秒值的掩码位。							
M	匹配。当此位设置为“0”时，在警报匹配中使用秒值。将此位设置为“1”会导致匹配电路忽略秒值。							
0x01	计时—世纪							
	D7	D6	D5	D4	D3	D2	D1	D0
	世纪 (x10)				世纪			
包含世纪的 BCD 值 低位半字节（四位）包含低位数字，范围为 0 到 9；高位半字节（四位）包含高位数字，范围为 0 到 9。寄存器的范围是 0 到 99 世纪。								
0x00	标志							
	D7	D6	D5	D4	D3	D2	D1	D0
	WDF	AF	PF	OSCF	BPF	CAL	W	R
WDF	看门狗定时器标志。当看门狗定时器允许在没有被用户复位的情况下可达到“0”时，该只读位设置为“1”。当标志寄存器被读或加电时，该位被清除为“0”。							
AF	警报标志。当时间和日期与储存在警报寄存器的值匹配且匹配位为“0”时，此只读位设置为“1”。当标志寄存器被读或加电时，警报标志清除。							
PF	断电标志。当电源下降到断电阀值 $V_{\text{SWITCH}}$ 时，只读位设置为“1”。当标志寄存器被读时，中断标志清除。							
OSCF	振荡器失败标志。如果振荡器启用而且在头 5 秒操作时间内未运行，加电时设置为“1”。这表示 RTC 备用电源中断而且时钟值不再有效。该位在电源循环后保持不变，从不被芯片内部清除。用户必须检查此条件并写入“0”以清除该标志。当用户复位 OSCF 标志位时，在 $t_{\text{RTCp}}$ 时间后位将更新。							
BPF	备用电源中断标志。如果备用电源（电池或电容）中断，加电时设置为“1”。备用电源中断条件由电压下降低于其各自最低指定电压确定。BPF 只能将数据保持到出现定义的备用电压 ( $V_{\text{DR}}$ ) 低电压电平。用户必须将此位复位以清除标志。用户复位 BPF 标志位后，在 $t_{\text{RTCp}}$ 时间后位将更新。							
CAL	校准模式。当设置为“1”时，INT 引脚输出 512 Hz 的方波。当设置为“0”时，INT 引脚恢复正常操作。该位优先于 SQ0/SQ1 和其他功能。加电时，该位默认为“0”（禁用）。							
W	写入启用：将“W”设置为“1”将冻结对 RTC 寄存器的更新。然后用户可写入 RTC 寄存器、警报寄存器、校准寄存器、中断寄存器和标志寄存器。如果时间已更改，将“W”设置为“0”会导致 RTC 寄存器中的内容传输到计时计数器中。此传输过程需要花费 $t_{\text{RTCp}}$ 的时间。加电时，此位默认为 0。							
R	读取启用。如果将“R”位设置为“1”，则会停止用户 RTC 寄存器中的时钟更新，以便在读取过程中不显示时钟更新。将“R”位设置为“0”以恢复对保持寄存器的时钟更新。设置该位不需要将“W”位设置为“1”。加电时，该位默认为 0。							

## 最大额定值

超过最大额定值可能会缩短组件的使用寿命。这些用户指导未经过测试。

存放温度 ..... -65 °C 至 +150 °C

最长累积存储时间

在 150°C 环境温度下 ..... 1000 个小时

在 85°C 环境温度下 ..... 20 年

最高结温 ..... 150 °C

相对于  $V_{SS}$  的  $V_{CC}$  供电电压

CY14C101I: ..... -0.5 V 至 +3.1 V

CY14B101I: ..... -0.5 V 至 +4.1 V

CY14E101I: ..... -0.5 V 至 +7.0 V

应用于 High Z 状态下的输出

的直流电压 ..... -0.5 V 至  $V_{CC} + 0.5 V$

输入电压 ..... -0.5 V 到  $V_{CC} + 0.5 V$

处于接地电位的

任何引脚的瞬变电压 ( $< 20 \text{ ns}$ ) ..... -2.0 V 至  $V_{CC} + 2.0 V$

封装功率散耗

能力 ( $T_A = 25^\circ\text{C}$ ) ..... 1.0 W

表面组装铅焊

温度 (3 秒) ..... +260 °C

直流输出电流 (一次输出 1, 持续 1 秒) ..... 15 mA

静电放电电压

(根据 MIL-STD-883, 方法 3015) ..... > 2001 V

栓锁电流 ..... > 140 mA

## 工作范围

产品	范围	环境温度	$V_{CC}$
CY14C101I	工业	-40 °C - +85 °C	2.4 V 至 2.6 V
CY14B101I			2.7 V 至 3.6 V
CY14E101I			4.5 V 至 5.5 V

## 直流电气特性

在工作范围内

参数	说明	测试条件	最小值	典型值 <sup>[9]</sup>	最大值	单位
$V_{CC}$	电源	CY14C101I	2.4	2.5	2.6	V
		CY14B101I	2.7	3.0	3.6	V
		CY14E101I	4.5	5.0	5.5	V
$I_{CC1}$	平均 $V_{CC}$ 电流	$f_{SCL} = 3.4 \text{ MHz}$ ; 无输出负载下获得的值 ( $I_{OUT} = 0 \text{ mA}$ )	—	—	1	mA
		$f_{SCL} = 1 \text{ MHz}$ ; 无输出负载下获得的值 ( $I_{OUT} = 0 \text{ mA}$ )	—	—	400	$\mu\text{A}$
		CY14E101I	—	—	450	mA
$I_{CC2}$	存储过程中的平均 $V_{CC}$ 电流	所有输入无需关注, $t_{STORE}$ 持续时间内的平均电流 $V_{CC} = \text{max}$	—	—	3	mA
$I_{CC4}$	自动存储循环期间的平均 $V_{CAP}$ 电流	所有输入无需关注, $t_{STORE}$ 持续时间内的平均电流	—	—	3	mA
$I_{SB}$	$V_{CC}$ 待机电流	$SCL \geq (V_{CC} - 0.2 \text{ V})$ 。 $V_{IN} \leq 0.2 \text{ V}$ 或 $\geq (V_{CC} - 0.2 \text{ V})$ 。 “W” 位设置为 “0”。非易失性循环完成后的待机电流强度。输入为静态。 $f_{SCL} = 0 \text{ MHz}$ 。	—	—	250	$\mu\text{A}$
$I_{ZZ}$	睡眠模式电流	寄存睡眠指令后的 $t_{SLEEP}$ 时间。所有输入为静态且在 CMOS 逻辑电平下配置。	—	—	8	$\mu\text{A}$
$I_{IX}^{[10]}$	每个 I/O 引脚的输入电流 (HSB 除外)	$0.1 V_{CC} < V_i < 0.9 V_{CCmax}$	-1	—	+1	$\mu\text{A}$
	每个 I/O 引脚的输入电流 (针对 HSB)		-100	—	+1	$\mu\text{A}$
$I_{OZ}$	输出漏电流		-1	—	+1	$\mu\text{A}$
$C_i$	每个 I/O 引脚的电容	在所有输入和输出信号引脚测得的电容和 $V_{SS}$ 。	—	—	7	pF

注

9. 典型值的温度为 25°C,  $V_{CC} = V_{CC(Typ)}$  非 100% 测试。

10. 不适用于 WP、A2 和 A1 引脚。

## 直流电气特性 (续)

在工作范围内

参数	说明	测试条件		最小值	典型值 <sup>[9]</sup>	最大值	单位
V <sub>IH</sub>	输入高电平电压			0.7 V <sub>CC</sub>	—	V <sub>CC</sub> + 0.5	V
V <sub>IL</sub>	输入低电平电压			− 0.5	—	0.3 V <sub>CC</sub>	V
V <sub>OL</sub>	输出低电平电压	I <sub>OL</sub> = 3 mA		0	—	0.4	V
R <sub>in</sub> <sup>[11]</sup>	输入电阻（WP、A2、A1）	针对 V <sub>IN</sub> = V <sub>IL</sub> (Max)		50	—	—	KΩ
		针对 V <sub>IN</sub> = V <sub>IH</sub> (Min)		1	—	—	MΩ
V <sub>hys</sub>	施密特触发器输入电压迟滞			0.05 V <sub>CC</sub>	—	—	V
V <sub>CAP</sub> <sup>[12]</sup>	存储电容	在 V <sub>CAP</sub> 引脚和 V <sub>SS</sub> 之间	CY14C101I	170	220	270	μF
			CY14B101I	42	47	180	μF
			CY14E101I				
V <sub>VCAP</sub> <sup>[13、14]</sup>	组件在 V <sub>CAP</sub> 引脚驱动的最大电压	V <sub>CC</sub> = Max	CY14C101I	—	—	V <sub>CC</sub>	V
			CY14B101I				
			CY14E101I	—	—	V <sub>CC</sub> − 0.5	V

## 数据保留与耐久性

在工作范围内

参数	说明	最小值	单位
$\text{DATA}_R$	数据保留	20	年
$\text{NV}_C$	非易失性存储操作	1,000	K

## 热阻

参数 <sup>[14]</sup>	说明	测试条件	16 引脚 SOIC	单位
$\Theta_{JA}$	热电阻 (结温)	根据 EIA/JESD51 的要求, 测试条件遵循测试热阻的标准测试方法和过程。	56.68	$^{\circ}\text{C}/\text{W}$
$\Theta_{JC}$	热电阻 (壳温)		32.11	$^{\circ}\text{C}/\text{W}$

### 注

11. 当输入电压低于  $V_{IL}$  时, 输入下拉电路较强 (50 K $\Omega$ )。当输入电压高于  $V_{IH}$  时, 输入下拉电路较弱 (1 M $\Omega$ )。
12.  $V_{CAP}$  的最小值可保证提供顺利完成自动存储所需要的电荷。 $V_{CAP}$  的最大值可保证  $V_{CAP}$  的电容在加电回读循环期间充电至最小电压, 以便紧急断电循环可以顺利完成自动存储。因此, 建议在规定的最小极限值和最大极限值内使用电容。有关  $V_{CAP}$  选项的详细信息, 请参见应用笔记 [AN43593](#)。
13. 当选择  $V_{CAP}$  电容的时候, 会提供  $V_{CAP}$  引脚上的最大电压 ( $V_{VCAP}$ ) 供指导之用。在工作温度范围之内的  $V_{CAP}$  电容的额定电压应高于  $V_{VCAP}$  电压。
14. 这些参数仅由设计保证, 未进行过测试。

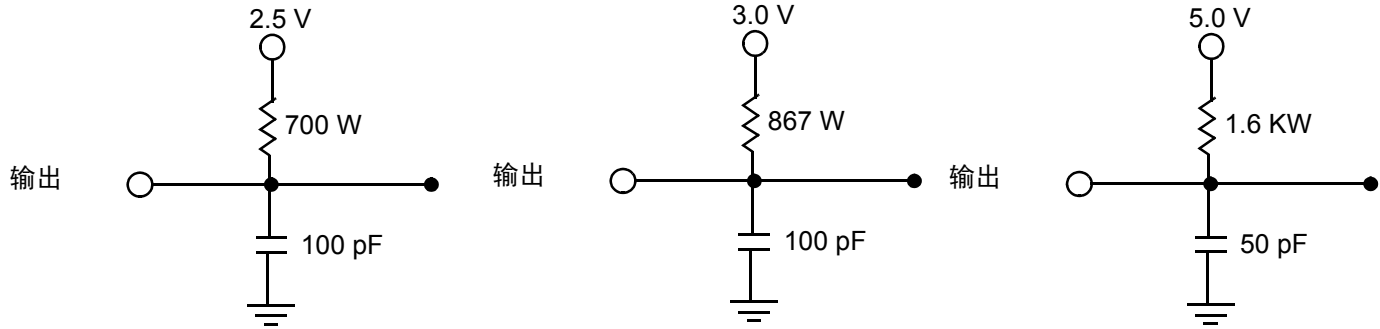
## 交流测试负载和波形

图 39. 交流测试负载和波形

对于 2.5 V (CY14C101I)

对于 3.0 V (CY14B101I)

对于 5.0 V (CY14E101I)



## 交流测试条件

说明	CY14C101I	CY14B101I	CY14E101I
输入脉冲电平	0 V 到 2.5 V	0 V 到 3 V	0 V 到 5 V
输出上升和下降时间 (10%–90%)	10 ns	10 ns	10 ns
输入和输出时序参考电平	1.25 V	1.5 V	2.5 V

## RTC 特性

在工作范围内

参数	说明	最小值	典型值	最大值	单位
$V_{RTCbat}$	RTC 电池引脚电压	1.8	–	3.6	V
$I_{BAK}^{[15]}$	RTC 备用电流	$T_A$ (Min)	–	–	0.45 $\mu A$
		25 °C	–	0.45	$\mu A$
		$T_A$ (Max)	–	0.60	$\mu A$
$V_{RTCcap}^{[16]}$	RTC 电容引脚电压	$T_A$ (Min)	1.6	–	3.6 V
		25 °C	1.5	3.0	3.6 V
		$T_A$ (Max)	1.4	–	3.6 V
$V_{BAKFAIL}$	备用电压故障阈值	1.8	–	2	V
$V_{DR}$	BPF 标志保留电压	1.6	–	–	V
$t_{OCS}$	RTC 振荡器启动时间	–	1	2	秒
$t_{RTCP}$	RTC 从“W”位末设置到“0”的处理时间。	–	–	1	ms
$R_{BKCHG}$	RTC 备用电容充电限流电阻	350	–	850	$\Omega$

### 注

15. 当  $V_{CC} < V_{SWITCH}$  时，由  $V_{RTCcap}$  或  $V_{RTCbat}$  抽取的电流。

16. 如果  $V_{RTCcap} > 0.5 V$ ，或没有电容连接到  $V_{RTCcap}$  引脚，振荡器将在  $t_{OCS}$  内启动。如果已连接备用电容且  $V_{RTCcap} < 0.5 V$ ，要启动振荡器，必须允许将电容充电至 0.5 V。

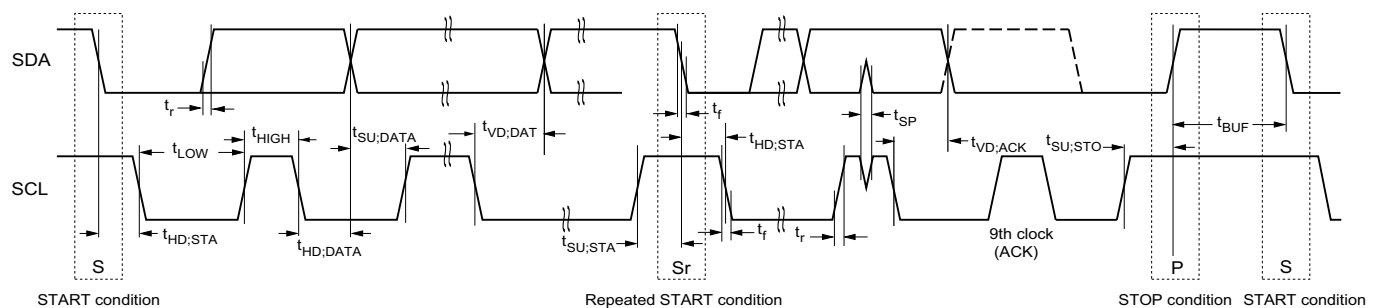
## 交流开关特性

在工作范围内<sup>[17]</sup>

参数	说明	3.4 MHz <sup>[18]</sup>		1 MHz <sup>[18]</sup>		400 kHz <sup>[18]</sup>		单位
		最小值	最大值	最小值	最大值	最小值	最大值	
$f_{SCL}$	时钟频率, SCL	—	3400	—	1000	—	400	kHz
$t_{SU; STA}$	重复 START 条件的建立时间	160	—	250	—	600	—	ns
$t_{HD; STA}$	START 条件的保持时间	160	—	250	—	600	—	ns
$t_{LOW}$	SCL 的低周期	160	—	500	—	1300	—	ns
$t_{HIGH}$	SCL 的高周期	60	—	260	—	600	—	ns
$t_{SU; DATA}$	数据输入的建立时间	10	—	100	—	100	—	ns
$t_{HD; DATA}$	数据保持时间 (输入 / 输出)	0	—	0	—	0	—	ns
$t_{DH}$	数据输出的保持时间	0	—	0	—	0	—	ns
$t_r^{[19]}$	SDA 和 SCL 的上升时间	—	80	—	120	—	300	ns
$t_f^{[19]}$	SDA 和 SCL 的下降时间	—	80	—	120	—	300	ns
$t_{SU; STO}$	STOP 条件的建立时间	160	—	250	—	600	—	ns
$t_{VD; DATA}$	数据输出有效时间	—	130	—	400	—	900	ns
$t_{VD; ACK}$	ACK 输出有效时间	—	130	—	400	—	900	ns
$t_{OF}^{[19]}$	从 $V_{IH\ min}$ 到 $V_{IL\ max}$ 的输出下降时间	—	80	—	120	—	300	ns
$t_{BUF}$	STOP 和下一个 START 条件之间的总线空闲时间	0.3	—	0.5	—	1.3	—	us
$t_{SP}$	必须被输入滤波器抑制的尖峰脉冲的宽度	—	10	—	50	—	50	ns

## 开关波形

图 40. 时序图



### 注

17. 测试条件采用等于或小于 10 ns 的信号跃迁时间、 $V_{CC}/2$  的时序参考电平、0 至  $V_{CC}$  (typ) 的输入脉冲电平以及指定  $I_{OL}$  的输出负载和图 39 中所示的负载电容。  
18. 总线负载电容 (Cb) 注意事项: 对于 I<sup>2</sup>C 时钟频率 (SCL) 100/400/1000 KHz,  $C_b < 500$  pF。对于 SCL 3.4 MHz,  $C_b < 100$  pF。  
19. 这些参数仅由设计保证, 未进行过测试。



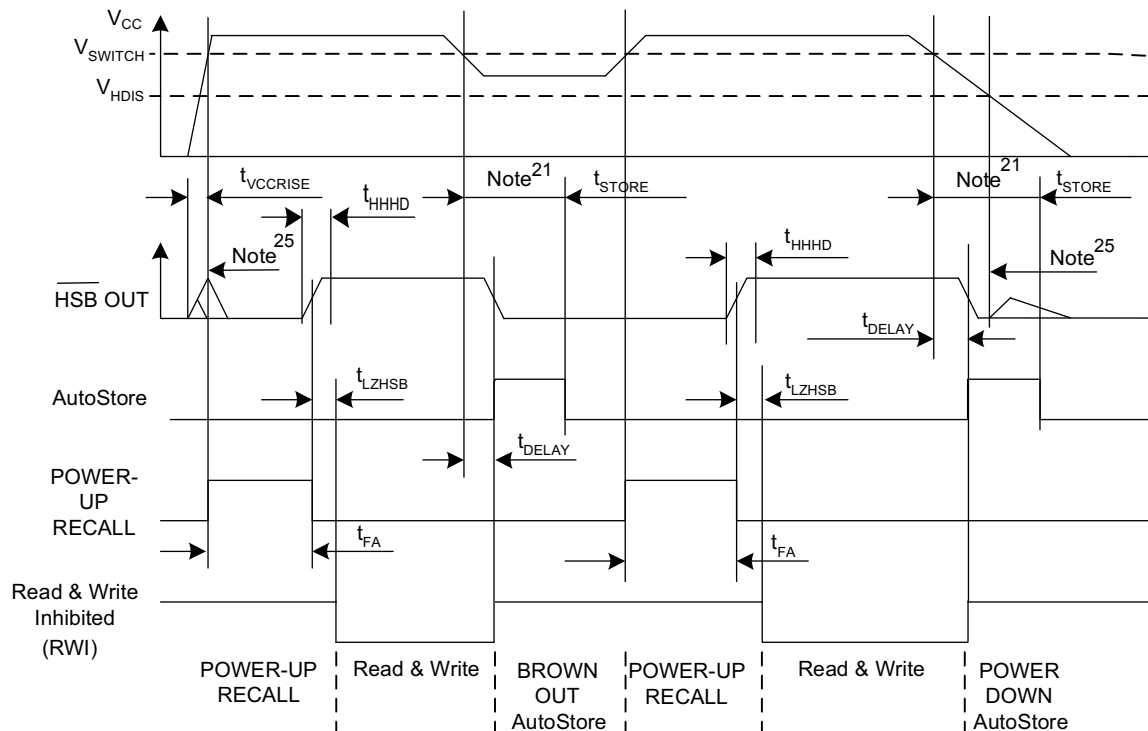
## nvSRAM 规范

在工作范围内

参数	说明		最小值	最大值	单位
$t_{FA}^{[20]}$	加电回读持续时间	CY14C101I	—	40	ms
		CY14B101I	—	20	ms
		CY14E101I	—	20	ms
$t_{STORE}^{[21]}$	存储循环持续时间		—	8	ms
$t_{DELAY}^{[22]}$	允许完成 SRAM 写入循环的时间		—	25	ns
$t_{VCCRISE}^{[23]}$	$V_{CC}$ 上升时间		150	—	$\mu s$
$V_{SWITCH}$	低电压触发电平	CY14C101I	—	2.35	V
		CY14B101I	—	2.65	V
		CY14E101I	—	4.40	V
$t_{LZHSB}^{[23]}$	HSB 高电平到 nvSRAM 激活时间。		—	5	$\mu s$
$V_{HDIS}^{[23]}$	HSB 输出禁用电压		—	1.9	V
$t_{HHHD}^{[23]}$	HSB 高电平有效时间		—	500	ns
$t_{WAKE}$	nvSRAM 从睡眠模式唤醒的时间	CY14C101I	—	40	ms
		CY14B101I	—	20	ms
		CY14E101I	—	20	ms
$t_{SLEEP}$	发出 SLEEP 指令后进入低功耗模式的时间		—	8	ms
$t_{SB}^{[23]}$	发出 STOP 条件后进入待机模式的时间		—	100	$\mu s$

## 开关波形

图 41. 自动存储或加电回读<sup>[24]</sup>



注

20.  $t_{FA}$  从  $V_{CC}$  超过  $V_{SWITCH}$  的那一刻开始。
21. 如果自上一个非易失性循环后未进行过 SRAM 写入，则不会发生自动存储或硬件存储。
22. 硬件存储和自动存储启动后，SRAM 写入操作在  $t_{DELAY}$  时间内持续启用。
23. 这些参数仅由设计保证，未进行过测试。
24. 在  $V_{CC}$  低于  $V_{SWITCH}$  的情况下，在存储、回读的过程中会忽略读写循环。
25. 在加电和断电期间，在通过外部电阻上拉 HSB 引脚时，HSB 发生短时脉冲。

## 软件控制的存储 / 回读循环

在工作范围内

参数	说明	CY14X101I		单位
		最小值	最大值	
$t_{\text{RECALL}}$	回读持续时间	—	600	$\mu\text{s}$
$t_{\text{SS}}^{[26, 27]}$	软件序列处理时间	—	500	$\mu\text{s}$

## 开关波形

图 42. 软件存储 / 回读循环

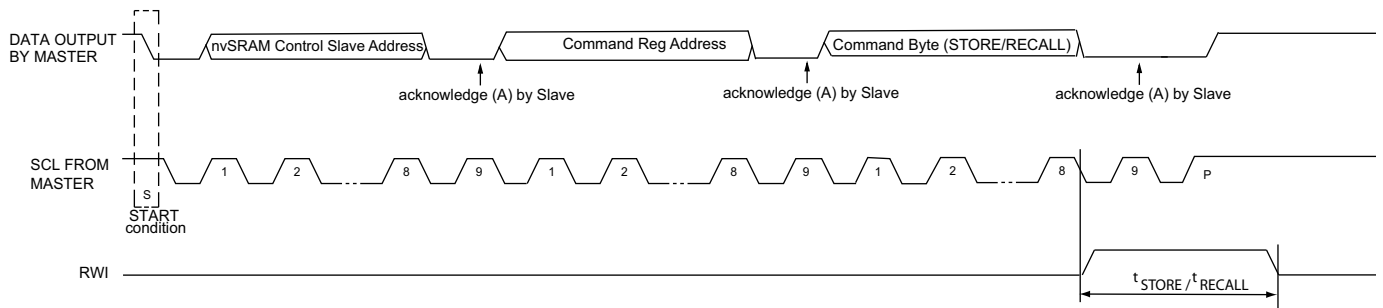
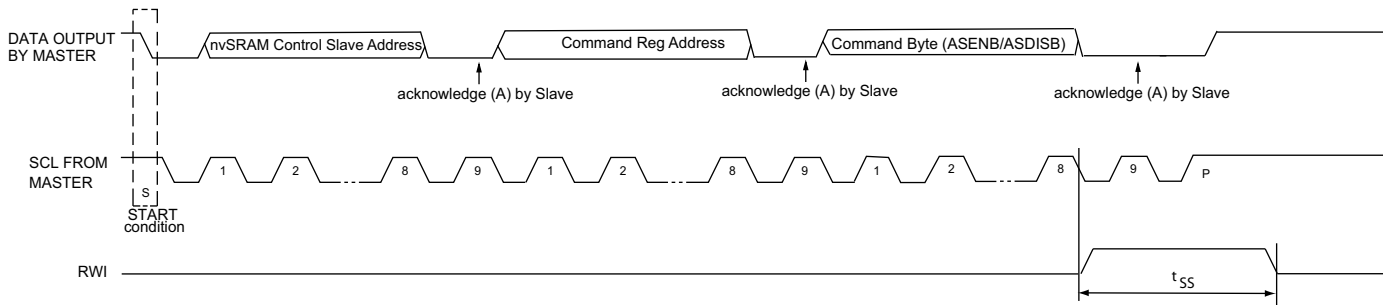


图 43. 自动存储启用 / 禁用循环



### 注

26. 这是执行软件序列命令所耗费的时间。V<sub>CC</sub> 电压必须保持高电平以保证有效地寄存命令。  
27. 存储和回读等命令会锁定 I/O，直到操作完成，这会进一步增加该时间。请参见特定的命令。

## 硬件存储循环

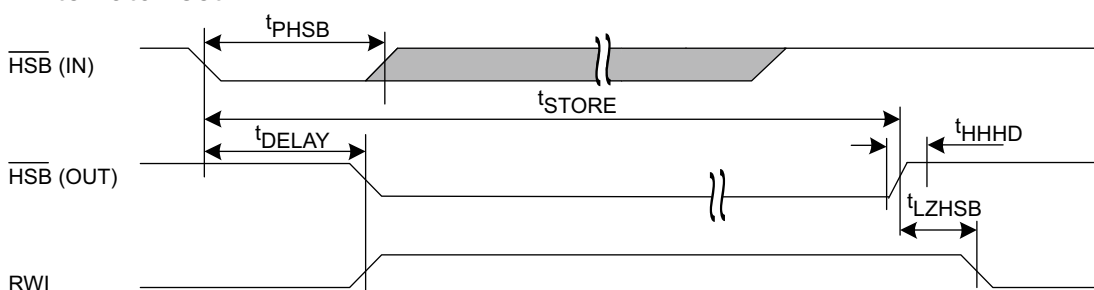
在工作范围内

参数	说明	CY14X101I		单位
		最小值	最大值	
$t_{PHSB}$	硬件存储脉冲宽度	15	—	ns

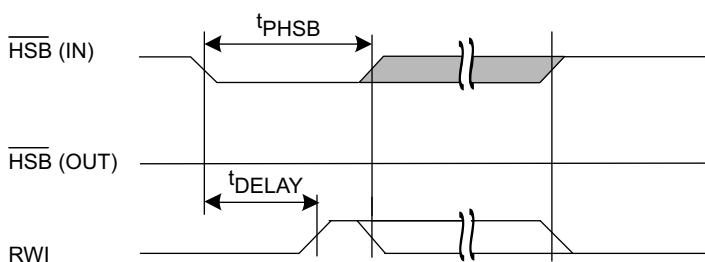
## 开关波形

图 44. 硬件存储循环<sup>[28]</sup>

### Write Latch set



### Write Latch not set



$\overline{HSB}$  pin is driven HIGH to  $V_{CC}$  only by Internal 100 K $\Omega$  resistor,  $\overline{HSB}$  driver is disabled  
SRAM is disabled as long as  $\overline{HSB}$  (IN) is driven LOW.

### 注

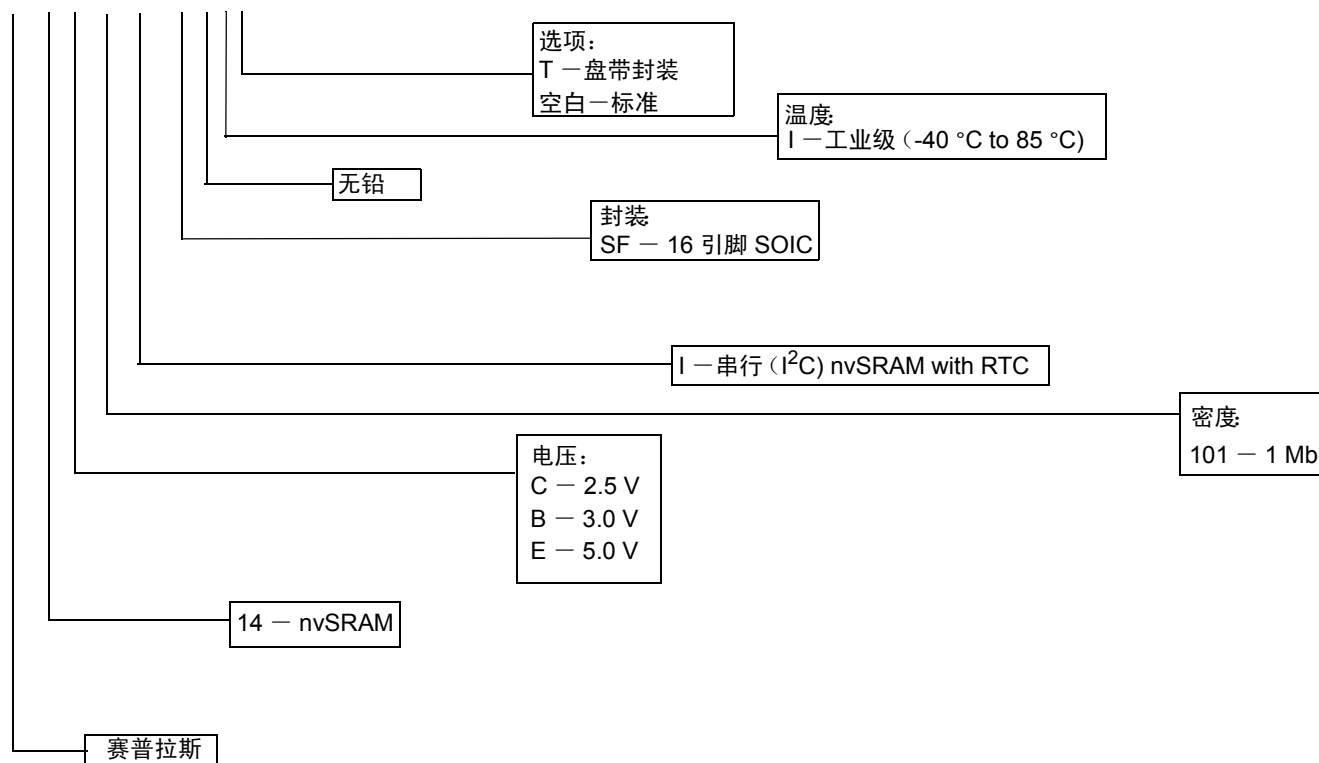
28. 如果自上一个非易失性循环后未进行过 SRAM 写入，则不会发生自动存储或硬件存储。

## 订购信息

订购代码	封装图	封装类型	工作范围
CY14B101I-SFXI	51-85022	16 引脚 SOIC	工业
CY14B101I-SFXIT			

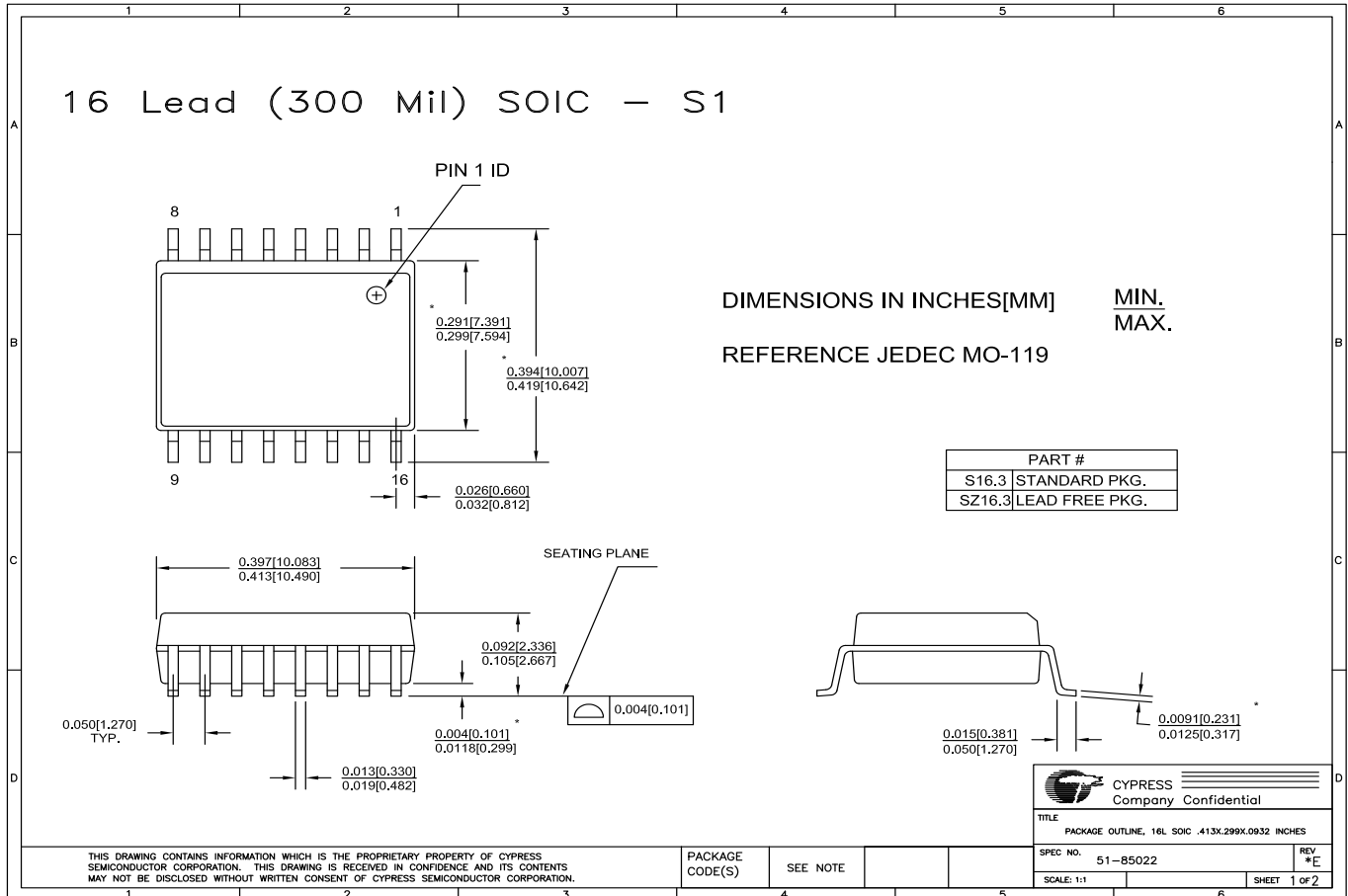
这些部件不含铅。

## 订购代码定义



## 封装图

图 45. 16 引脚 SOIC (0.413 × 0.299 × 0.0932 英寸) 封装外形, 51-85022



## 缩略语

缩略语	说明
ACK	确认
BCD	二进制十进数
CMOS	互补金属氧化物半导体
CRC	循环冗余校验
EIA	电子工业联盟
I <sup>2</sup> C	内部集成电路
I/O	输入 / 输出
JEDEC	联合电子组件工程委员会
LSB	最低有效位
MSB	最高有效位
nvSRAM	非易失性静态随机存取存储器
NACK	未确认
OSCF	振荡器失败标志
RoHS	有害物质限制
RTC	实时时钟
R/W	读 / 写
RWI	禁止读和写
SCL	串行时钟线
SDA	串行数据访问
SNL	序列号锁定
SOIC	小外形集成电路
SRAM	静态随机存取存储器
WP	写保护

## 文档规范

### 测量单位

符号	测量单位
°C	摄氏度
F	法拉
Hz	赫兹
kHz	千赫兹
kΩ	千欧
Mbit	兆位
MHz	兆赫兹
MΩ	兆欧
μA	微安
μF	微法
μs	微秒
mA	毫安
ms	毫秒
ns	纳秒
Ω	欧姆
%	百分比
pF	皮法
Sec	秒
V	伏特
W	瓦特



## 文档修订记录页

文档标题: CY14C101I、CY14B101I、CY14E101I, 具有实时时钟功能的 1-Mbit (128 K × 8) 串行 (I <sup>2</sup> C) nvSRAM				
文档编号: 001-87305				
修订版本	ECN 编号	提交日期	变更方	变更说明
**	3980699	04/24/2013	HLIN	本文档版本号为 Rev**, 译自英文版 001-54391 Rev*I

## 销售、解决方案和法律信息

### 全球销售和设计支持

赛普拉斯公司拥有一个由办事处、解决方案中心、工厂代表和经销商组成的全球性网络。要找到距您最近的办事处，请访问[赛普拉斯所在地](#)。

### 产品

汽车用产品	<a href="http://cypress.com/go/automotive">cypress.com/go/automotive</a>
时钟与缓冲器	<a href="http://cypress.com/go/clocks">cypress.com/go/clocks</a>
接口	<a href="http://cypress.com/go/interface">cypress.com/go/interface</a>
照明与电源控制	<a href="http://cypress.com/go/powerpsoc">cypress.com/go/powerpsoc</a> <a href="http://cypress.com/go/plc">cypress.com/go/plc</a>
存储器	<a href="http://cypress.com/go/memory">cypress.com/go/memory</a>
光学与图像传感器	<a href="http://cypress.com/go/image">cypress.com/go/image</a>
PSoC	<a href="http://cypress.com/go/psoc">cypress.com/go/psoc</a>
触摸感应产品	<a href="http://cypress.com/go/touch">cypress.com/go/touch</a>
USB 控制器	<a href="http://cypress.com/go/USB">cypress.com/go/USB</a>
无线 /RF 产品	<a href="http://cypress.com/go/wireless">cypress.com/go/wireless</a>

### PSoC 解决方案

[psoc.cypress.com/solutions](http://psoc.cypress.com/solutions)  
PSoC 1 | PSoC 3 | PSoC 5

© 赛普拉斯半导体公司，2013。此处所包含的信息可能会随时更改，恕不另行通知。除赛普拉斯产品的内嵌电路之外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不根据专利或其他权利以明示或暗示的方式授予任何许可。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯产品不保证能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

所有源代码（软件和 / 或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和 / 或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定的用途之外，未经赛普拉斯的明确书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对此材料提供任何类型的明示或暗示保证，包括（但不限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不另行通知的情况下对此处所述材料进行更改的权利。赛普拉斯不在此处所述之任何产品或电路的应用或使用承担任何责任。对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

产品使用可能受适用的赛普拉斯软件许可协议限制。