

**ANALOG
DEVICES**

**低功耗、精密模拟、微控制器、
双 Σ - Δ ADC、Flash/EE、ARM7TDMI**

ADuC7060/ADuC7061

特性

模拟输入/输出

- 双(24位)ADC
- 单端和差分输入
- 可编程ADC输出速率(4 Hz至8 kHz)
- 可编程数字滤波器
- 内置系统校准
- 低功耗工作模式

主(24位)ADC通道

- 两个差分对或四个单端通道
- PGA(1至512)输入级
- 可选输入范围: ± 2.34 mV至 ± 1.2 V
- 30 nV均方根噪声

辅助(24位)ADC: 四个差分对或七个单端通道

- 片内精密基准电压源(± 10 ppm/ $^{\circ}$ C)

可编程传感器激励电流源

- 200 μ A至2 mA电流源范围

单14位电压输出DAC

微控制器

- 16位/32位RISC架构ARM7TDMI内核
- JTAG端口支持代码下载和调试
- 多个时钟选项

内存

- 32 kB (16 kB \times 16) Flash/EE存储器, 包括2 kB内核
- 4 kB (1 kB \times 32) SRAM

工具

- 在线下载, 基于JTAG调试
- 低成本QuickStart开发系统™

通信接口

SPI接口(5 Mbps)

- 4字节接收和发送FIFO

UART串行I/O和I²C(主/从)

片内外设

- 包括4个通用(捕捉)定时器
- 唤醒定时器
- 看门狗定时器

用于FIQ和IRQ的矢量中断控制器

- 每类中断支持8种优先级
- 边沿或电平中断外部引脚输入

16位, 6通道PWM

通用输入/输出

- 最多14个完全兼容3.3V的GPIO引脚

电源

- AVDD/DVDD额定电压2.5 V ($\pm 5\%$)

主动模式: 2.74 mA (@ 640 kHz, ADC0主动)

10 mA (@ 10.24 MHz, 两个ADC主动)

封装和温度范围

- 工作温度范围: -40° C至 $+125^{\circ}$ C完全性能指标保证
- 32引脚LFCSP封装(5 mm \times 5 mm)
- 48引脚LFCSP和LQFP

衍生产品

- 32引脚LFCSP(ADuC7061)
- 48引脚LQFP和48引脚LFCSP(ADuC7060)

应用

- 工业自动化和过程控制
- 智能、精密传感系统、4 mA至20 mA
- 环路智能传感器

概述

ADuC7060/ADuC7061系列是完全集成的8 kSPS、24位数据采集系统, 在单芯片内集成高性能多通道 Σ - Δ 型模数转换器(ADC)、16位/32位ARM7TDMI® MCU和Flash/EE存储器。

它由内置两个差分对或四个单端通道的主ADC和内置七个通道的辅助ADC组成, 可以工作在单端模式或差分输入模式下。片内集成一个单通道缓冲电压输出DAC。通过编程可以将DAC输出范围设置为四种电压范围之一。

这些器件通过一个片内振荡器和锁相环(PLL)产生最多10.24 MHz的内部高频时钟信号。微控制器内核为ARM7TD-MI、16位/32位RISC机器, 峰值性能高达10 MIPS; 片内集成4 kB SRAM和32 kB非易失性Flash/EE存储器。ARM7TD-MI内核将所有存储器和寄存器视为一个线性阵列。

ADuC7060/ADuC7061包括四个定时器。定时器1为唤醒定时器, 具有使器件离开省电模式的功能。定时器2可配置为看门狗定时器。同时提供了一个带6个输出通道的16位PWM。ADuC7060/ADuC7061内置一个高级中断控制器。该矢量中断控制器(VIC)可以为每个中断分配一个优先级。它还支持嵌套中断, 每个IRQ和FIQ最多允许8级嵌套。如果将IRQ和FIQ中断源合并, 则可以支持总计16级嵌套中断。片内出厂固件支持通过UART串行接口端口进行在线下载, 同时也支持通过JTAG接口进行非介入仿真。这些器件采用2.375 V至2.625 V电源供电, 工作温度范围为 -40° C至 $+125^{\circ}$ C工业温度范围。

Rev. E

[Document Feedback](#)

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 ©2009–2014 Analog Devices, Inc. All rights reserved.
[Technical Support](#) www.analog.com

ADI中文数据手册是英文版数据手册的译文, 敬请谅解翻译中可能存在的语言组织或翻译错误, ADI不对翻译中存在的差异或由此产生的错误负责。如需确认任何词语的准确性, 请参考ADI提供的最新英文版数据手册。

目录

| | | | |
|-------------------|----|--------------------------------|-----|
| 特性..... | 1 | 寄存器接口..... | 57 |
| 应用..... | 1 | 使用DAC..... | 58 |
| 概述..... | 1 | 非易失Flash/EE存储器..... | 59 |
| 修订历史..... | 3 | Flash/EE存储器可靠性..... | 59 |
| 功能框图..... | 5 | 编程..... | 59 |
| 技术规格..... | 6 | 处理器相关外设..... | 60 |
| 电气规格..... | 6 | 中断系统..... | 60 |
| 时序规格..... | 11 | IRQ..... | 60 |
| 绝对最大额定值..... | 15 | 快速中断请求(FIQ)..... | 61 |
| ESD警告..... | 15 | 可编程中断..... | 62 |
| 引脚配置和功能描述..... | 16 | 矢量中断控制器(VIC)..... | 62 |
| 术语..... | 21 | VIC寄存器..... | 62 |
| ARM7TDMI内核概览..... | 22 | 定时器..... | 67 |
| Thumb模式(T)..... | 22 | 时:分:秒:1/28格式..... | 67 |
| 乘法器(M)..... | 22 | 定时器0..... | 68 |
| 嵌入式ICE (I)..... | 22 | 定时器1或唤醒定时器..... | 70 |
| ARM寄存器..... | 22 | 定时器2或看门狗定时器..... | 72 |
| 中断延迟..... | 23 | 定时器3..... | 74 |
| 存储器结构..... | 23 | 脉宽调制器..... | 76 |
| Flash/EE控制接口..... | 24 | 脉宽调制器概述..... | 76 |
| 存储器映射寄存器..... | 28 | UART串行接口..... | 81 |
| 完整的寄存器列表..... | 29 | 波特率生成..... | 81 |
| 复位..... | 34 | UART寄存器定义..... | 81 |
| 振荡器、锁相环和电源控制..... | 35 | I ² C..... | 87 |
| 时钟系统..... | 35 | I ² C功能的外部引脚配置..... | 87 |
| 电源控制系统..... | 35 | 串行时钟生成..... | 88 |
| ADC电路信息..... | 39 | I ² C总线地址..... | 88 |
| 基准电压源..... | 40 | I ² C寄存器..... | 88 |
| 诊断电流源..... | 40 | 串行外设接口..... | 97 |
| Sinc3滤波器..... | 41 | MISO(主机输入、从机输出)引脚..... | 97 |
| ADC斩波..... | 41 | MOSI(主机输出、从机输入)引脚..... | 97 |
| 可编程增益放大器..... | 41 | SCLK(串行时钟输入/输出)引脚..... | 97 |
| 激励源..... | 41 | 从机选择(P0.0/)输入引脚..... | 97 |
| ADC低功耗模式..... | 41 | SPI功能的外部引脚配置..... | 97 |
| ADC比较器和累加器..... | 42 | SPI寄存器..... | 98 |
| 温度传感器..... | 42 | 通用I/O..... | 102 |
| ADC寄存器接口..... | 42 | GPxCON寄存器..... | 102 |
| 应用电路示例..... | 55 | GPxDAT寄存器..... | 103 |
| DAC外设..... | 57 | GPxSET寄存器..... | 103 |
| DAC..... | 57 | GPxCLR寄存器..... | 103 |

| | |
|-----------------|-----|
| GPxPAR寄存器 | 103 |
| 硬件设计考虑 | 105 |
| 电源 | 105 |

修订历史

2014年10月—修订版D至修订版E

| | |
|---|-----|
| 更改表1 | 6 |
| FEESIGN更改为FEESIG(通篇) | 25 |
| FEEHIDE更改为FEEHID(通篇) | 26 |
| 更改表36 | 40 |
| 更改“Sinc3滤波器”部分 | 41 |
| 更改表43 | 46 |
| 更改“ADC滤波器寄存器”部分 | 48 |
| 更改表94 | 85 |
| 更改I ² C部分 | 87 |
| 更改表97 | 89 |
| 表98中寄存器I2CMSTA的位7从I2CMNA 更改为I2CMND | 90 |
| 更改表104 | 93 |
| 更改表105 | 94 |
| 更新“外形尺寸”部分的图31 | 106 |

2012年4月—修订版C至修订版D

| | |
|--|-----|
| 更改表1 | 6 |
| 更改表7 | 14 |
| 更改表16 | 25 |
| 更改“批量擦除命令执行序列”部分 | 26 |
| 更改表19 | 29 |
| 更改“电源和时钟控制寄存器”部分 | 35 |
| 更改图20 | 55 |
| 更改表63中的位5 | 57 |
| 更改“定时器”部分；增加“时:分:秒:1/128格式”部分 和表79；重新排序 | 67 |
| 更改“定时器1或唤醒定时器”部分 | 70 |
| 更改“定时器2载入寄存器”部分和“定时器2数值寄存器” 部分 | 71 |
| 增加表108 | 98 |
| 更新“外形尺寸” | 105 |

2011年5月—修订版B至修订版C

| | |
|---|----|
| 更改图1 | 4 |
| 更改表1 | 6 |
| 增加“临时保护”部分和“加密永久性保护”部分 | 25 |
| 增加“永久性保护”和“写入软件保护密钥和设置永久性保护的 序列”部分 | 26 |
| 更改“电源控制系统”部分 | 35 |
| 更改表43中的位9:6 | 45 |

| | |
|------------------------------|-----|
| 更改“主通道ADC数据寄存器”部分和表49 | 50 |
| 更改“IRQEN”部分和“IRQCLR”部分 | 59 |
| 外形尺寸 | 106 |
| 订购指南 | 107 |

| | |
|------------------------|-----|
| 更改“定时器1或唤醒定时器”部分 | 69 |
| 更改表108 | 101 |

2010年2月—修订版A至修订版B

| | |
|-------------------------------|-----|
| 更改特性部分 | 1 |
| 更改表1 | 4 |
| 更改数字I/O电压至DGND参数 | 14 |
| 更改引脚19、引脚20和引脚45的描述(表8) | 16 |
| 更改引脚13、引脚14和引脚29的描述(表9) | 18 |
| 更改表14的位8 | 23 |
| 更改表20 | 28 |
| 更改电源控制系统部分 | 34 |
| 增加表32 | 35 |
| 更改表34中尾注2和尾注3 | 36 |
| 更改表42 | 42 |
| 更改表43中位12和位[3:0] | 44 |
| 更改表44的位12 | 45 |
| 更改表45的尾注2 | 47 |
| 更改表63的位5 | 55 |
| 更改串行下载(在线编程)部分 | 57 |
| 更改优先级寄存器部分 | 61 |
| 更改GPxPAR寄存器部分 | 101 |

2009年6月—修订版0至修订版A

| | |
|------------------------|----|
| 增加ADuC7061 | 通用 |
| 增加新封装CP-32-4 | 通用 |
| 更改特性部分 | 1 |
| 更改概述部分 | 1 |
| 更改图1 | 4 |
| 更改表1 | 7 |
| 更改表2的尾注 | 10 |
| 更改表3和表4的尾注 | 11 |
| 更改表5的尾注 | 12 |
| 更改表6的尾注 | 13 |
| 更改图7和表8 | 15 |
| 增加图8和表9，重新排序 | 18 |
| 更改Flash/EE控制接口部分 | 23 |
| 更改表15的编码0x04描述 | 24 |
| 更改表16的位31描述 | 25 |
| 更改表17 | 27 |
| 更改表19 T0CLRI和表20 | 28 |
| 更改表21的尾注 | 29 |
| 更改表25的SPITX默认值 | 30 |

ADuC7060/ADuC7061

| | | | |
|---|----|----------------------------|-----|
| 更改外部时钟选择部分..... | 33 | 更改图24和表79..... | 68 |
| 更改ADC电路信息部分..... | 36 | 更改定时器2接口部分和图25..... | 69 |
| 更改表35的列标题..... | 37 | 更改定时器3捕捉寄存器部分..... | 71 |
| 更改表39的位6描述..... | 40 | 更改表81的位[16:12]描述..... | 72 |
| 更改表43的位12描述..... | 44 | 更改表82和图26的“脉宽调制器概述”部分..... | 73 |
| 更改“主通道ADC数据寄存器”部分和“辅助通道ADC数据寄存器”部分..... | 48 | 更改表84的列标题..... | 75 |
| 更改表59和图17..... | 51 | 更改表92..... | 82 |
| 更改使用DAC部分..... | 55 | 更改表102的位1..... | 90 |
| 更改“非易失Flash/EE存储器”部分和“编程”部分..... | 56 | 更改表105的位11描述..... | 95 |
| 更改矢量中断控制器(VIC)部分..... | 59 | 更改表106的SPIMDE位描述..... | 97 |
| 更改优先级寄存器部分..... | 60 | 更新外形尺寸..... | 103 |
| 更改表73..... | 61 | 更改订购指南..... | 104 |
| 更改图23..... | 65 | | |
| 更改表78..... | 66 | | |

2009年4月—版本0：初始版

功能框图

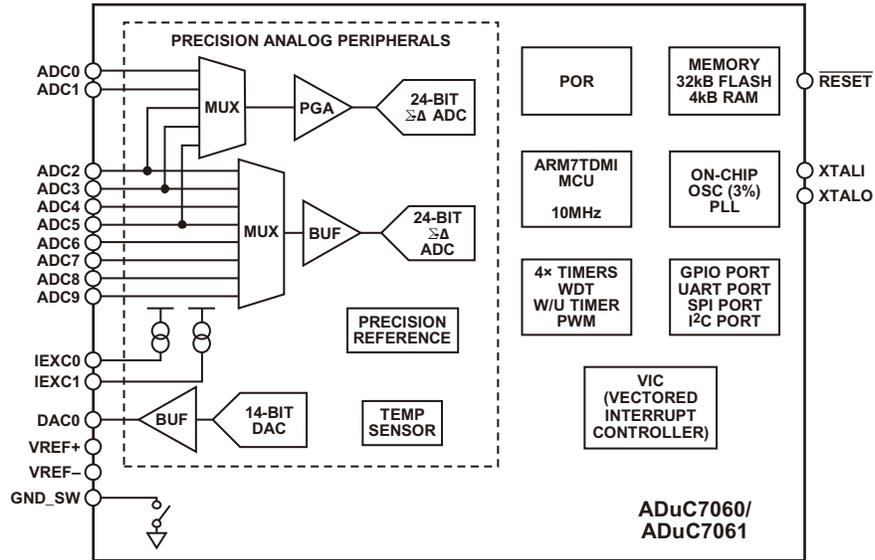


图1.

07075-001

ADuC7060/ADuC7061

技术规格

电气规格

除非另有说明, $V_{DD} = 2.5 V \pm 5\%$, $V_{REF+} = 1.2 V$, $V_{REF-} = GND$, $f_{CORE} = 10.24 MHz$, 采用外部32.768 kHz时钟晶体或片内振荡器驱动, 所有规格均相对于 $T_A = -40^\circ C$ 至 $+125^\circ C$ 。输出噪声规格可参考表36(主ADC)和表38(ADC辅助通道)。

表1. ADuC7060/ADuC7061规格

| 参数 | 测试条件/注释 | 最小值 | 典型值 | 最大值 | 单位 |
|---------------------------|--|------|--------------|------|------------------|
| ADC技术规格 | 对于所有ADC规格, 除非另有说明 否则假定处于正常工作模式 | | | | |
| 转换速率 ¹ | 斩波关闭, ADC正常工作模式 | 50 | | 8000 | Hz |
| | 斩波开启, ADC正常工作模式 | 4 | | 2600 | Hz |
| | 斩波开启, ADC低功耗模式 | 1 | | 650 | Hz |
| 主通道 | | | | | |
| 无失码 ¹ | 斩波关闭($f_{ADC} \leq 1 kHz$) | 24 | | | 位 |
| | 斩波开启($f_{ADC} \leq 666 kHz$) | 24 | | | 位 |
| 积分非线性(INL) ^{1,2} | 增益 = 4 | | ± 15 | | FSR的ppm |
| 失调误差 ^{3,4} | 斩波关闭, 校准之后, 失调误差 与选定的编程增益和更新速率所 对应的噪声相当。 | -27 | ± 8 | +27 | μV |
| 失调误差 ^{1,3,4} | 斩波开启 | -2.7 | ± 0.5 | +2.7 | |
| 失调误差漂移与温度的关系 ⁵ | 斩波关闭(增益 ≤ 64) | | 650/PGA_GAIN | | $\mu V/^\circ C$ |
| | 斩波开启(增益 ≤ 64) | | 10 | | $nV/^\circ C$ |
| 满量程误差 ^{1,6,7,8} | 正常模式 | -1 | ± 0.5 | +1 | mV |
| 满量程误差 ^{6,8} | 低功耗模式 | -2 | ± 1.0 | +2 | mV |
| 增益漂移与温度的关系 ⁹ | | | 5 | | ppm/ $^\circ C$ |
| PGA增益不匹配误差 | | | ± 0.1 | | % |
| 电源抑制 ¹ | 斩波开启, ADC = 1 V(增益 = 1) | | 65 | | dB |
| | 斩波开启, ADC = 7.8 mV(增益 = 128) | 84.7 | 113 | | dB |
| | 斩波关闭, ADC = 1 V(增益 = 1) | 56 | 65 | | dB |
| 辅助通道 | | | | | |
| 无失码 ¹ | 斩波关闭($f_{ADC} \leq 1 kHz$) | 24 | | | 位 |
| | 斩波开启($f_{ADC} \leq 666 kHz$) | 24 | | | 位 |
| 积分非线性(INL) ¹ | | | ± 15 | | FSR的ppm |
| 失调误差 ⁴ | 斩波关闭 | -120 | ± 30 | +100 | |
| 失调误差 ^{1,4} | 斩波开启 | -1.5 | ± 0.5 | +3.2 | μV |
| 失调误差漂移与温度的关系 ⁵ | 斩波关闭 | | 200 | | $\mu V/^\circ C$ |
| | 斩波开启 | | 10 | | $nV/^\circ C$ |
| 满量程误差 ^{1,6,7,8} | 正常模式 | -1 | ± 0.5 | +1 | mV |
| 满量程误差 ^{1,6,8} | 低功耗模式 | -2 | ± 1.0 | +2 | mV |
| 增益漂移与温度的关系 ⁹ | | | 3 | | ppm/ $^\circ C$ |
| 电源抑制 ¹ | 斩波开启, ADC = 1 V | 55 | 65 | | dB |
| | 斩波关闭, ADC = 1 V | 53 | 65 | | dB |

ADuC7060/ADuC7061

| 参数 | 测试条件/注释 | 最小值 | 典型值 | 最大值 | 单位 |
|---------------------------------------|--|------|----------|-----------------|-----------------------|
| ADC技术规格: 模拟输入 | 内部 $V_{REF} = 1.2\text{ V}$ | | | | |
| 主通道 | | | | | |
| 绝对输入电压范围 | 同时适用于VIN+和VIN- | 0.1 | | $V_{DD} - 0.7$ | V |
| 输入电压范围 | 增益 = 1 ¹ | 0 | | 1.2 | V |
| (AIN+和AIN-之间的差分电压) | 增益 = 2 ¹ | 0 | | 600 | mV |
| | 增益 = 4 ¹ | 0 | | 300 | mV |
| | 增益 = 8 ¹ | 0 | | 150 | mV |
| | 增益 = 16 ¹ | 0 | | 75 | mV |
| | 增益 = 32 ¹ | 0 | | 37.5 | mV |
| | 增益 = 64 ¹ | 0 | | 18.75 | mV |
| | 增益 = 128 ¹ | 0 | | 9.375 | mV |
| 共模电压 V_{CM} ¹⁰ | $V_{CM} = (AIN(+) + AIN(-))/2$, 增益 = 4至128 | 0.5 | | | V |
| 输入漏电流 ¹ | ADC0和ADC1 | | 10 | 18 ¹ | nA |
| | ADC2、ADC3、ADC4和ADC5 | | 15 | 30 ¹ | nA |
| | ADC6、ADC7、ADC8和ADC9、 VREF+、VREF- 在ADC不工作时测量 | | 15 | 25 ¹ | nA |
| 共模抑制DC ¹ | | | | | |
| ADC输入 | ADC = 7.8 mV | 113 | | | dB |
| | ADC = 1 V ¹ | | 95 | | dB |
| 共模抑制50 Hz/60 Hz ¹ | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz和50 Hz 更新速率, 斩波开启 | 95 | | | dB |
| | ADC = 7.8 mV, 范围 $\pm 20\text{ mV}$ | 90 | | | dB |
| | ADC = 1 V, 范围 $\pm 1.2\text{ V}$ | | | | dB |
| 串模干扰抑制50 Hz/60 Hz ¹ | | | | | |
| ADC输入 | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz f _{ADC} , 斩波开启 | 75 | | | dB |
| | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz f _{ADC} , 斩波关闭 | 67 | | | dB |
| 辅助通道 | | | | | |
| 绝对输入电压范围 ¹ | 缓冲使能 | 0.1 | | $AV_{DD} - 0.1$ | V |
| | 缓冲禁用 | AGND | | AV_{DD} | V |
| 输入电压范围 | 范围-基准电压源 | 0 | | 1.2 | V |
| 共模抑制DC ¹ | | | | | |
| ADC输入 | ADC = 1 V ¹ | | 87 | | dB |
| 共模抑制50 Hz/60 Hz ¹ | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz和50 Hz 更新速率, 斩波开启 | 90 | | | dB |
| | ADC = 1 V, 范围 $\pm 1.2\text{ V}$ | | | | dB |
| 串模干扰抑制50 Hz/60 Hz ¹ | | | | | |
| ADC输入 | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz f _{ADC} , 斩波开启 | 75 | | | dB |
| | 50 Hz/60 Hz $\pm 1\text{ Hz}$, 16.6 Hz f _{ADC} , 斩波关闭 | 67 | | | dB |
| 基准电压源 | | | | | |
| ADC精密基准电压源 | | | | | |
| 内部 V_{REF} | | | 1.2 | | V |
| 初始精度 | 在 $T_A = 25^\circ\text{C}$ 时测定 | -0.1 | | +0.1 | % |
| 基准电压源温度系数 (Tempco) ^{1,11} | | -20 | ± 10 | +20 | ppm/ $^\circ\text{C}$ |
| 电源抑制 ¹ | | | 70 | | dB |
| 外部基准电压输入范围 ¹² | | 0.1 | | AV_{DD} | V |
| V_{REF} 二分频初始误差 ¹ | | | 0.1 | | % |

ADuC7060/ADuC7061

| 参数 | 测试条件/注释 | 最小值 | 典型值 | 最大值 | 单位 |
|---|--|--------|--|---|--|
| DAC通道规格 电压范围 | $R_L = 5\text{ k}\Omega, C_L = 100\text{ pF}$ | 0 0 | | V_{REF} $AVDD - 0.2$ | V V |
| DAC 12位模式 直流特性 ¹³ 分辨率 相对精度 微分非线性(DNL) 失调误差 增益误差 增益误差失配 | 保证单调性 1.2 V内部基准电压 V_{REF} 范围(基准电压 = 1.2 V) AVDD范围 | 12 | ± 2 ± 0.2 ± 2 | ± 1 ± 15 ± 1 ± 1 | 位 LSB LSB mV % % DAC量程的百分比 |
| DAC 16位模式 ¹ 直流特性 ¹⁴ 分辨率 相对精度 微分非线性(DNL) 失调误差 增益误差 增益误差失配 | 仅14位单调性 14位分辨率 保证单调性(14位) 1.2 V内部基准电压 V_{REF} 范围(基准电压 = 1.2 V) AVDD范围 | 14 | ± 3 ± 0.5 ± 2 ± 1 ± 1 0.1 | ± 1 ± 15 | 位 LSB LSB mV % % DAC量程的百分比 |
| DAC交流特性 输出电压建立时间 数模转换毛刺能量 | 主进位1 LSB变化(DAC0DAT寄存器中同时变化的最大位数) | | 10 ± 20 | | μs nV-sec |
| 温度传感器 ^{1, 15} 精度 0°C时电压输出 电压温度系数 热阻 | 用户校准后 MCU处于省电或待机模式 典型值 典型值 48引脚LFCSP封装 48引脚LQFP 32引脚LFCSP封装 | | ± 4 96 0.28 27 55 30 | | °C mV mV/°C °C/W °C/W °C/W |
| 接地开关 ¹ R_{ON} 输入漏电流 | | | 10 15 | | Ω nA |
| 上电复位(POR) POR触发电平 ¹ $\overline{\text{RESET}}$ POR复位超时 | 参见DVDD引脚的电压 上电电平 关断电平 1.8V和2.25V之间的最大电源斜坡; POR触发后。DVDD必须在此时间限制内达到2.25 V。 | | 2.0 2.25 | 128 | V V ms |

ADuC7060/ADuC7061

| 参数 | 测试条件/注释 | 最小值 | 典型值 | 最大值 | 单位 |
|--------------------------|--|--------------|--------|--------------|--------|
| 激励电流源 | | | | | |
| 输出电流 | 各电流源可用 | 200 | 1000 | | μA |
| 初始容差(25°C) | | | ±5 | | % |
| 漂移 ¹ | | | 0.06 | | %/°C |
| 25°C时初始电流匹配 | 在两个电流源之间匹配 | | ±0.5 | | % |
| 漂移匹配 ¹ | | | 20 | | ppm/°C |
| 电压调整率(AVDD) ¹ | AVDD = 2.5 V ± 5% | | 0.2 | | %/V |
| 输出电压电压 ¹ | | AGND – 30 mV | | AVDD – 0.7 V | V |
| 看门狗定时器(WDT) | | | | | |
| 超时周期 ¹ | 32.768 kHz时钟, 256预分频 | 0.008 | | 512 | sec |
| 超时步幅 | | | 7.8 | | ms |
| FLASH/EE存储器 ¹ | | | | | |
| 耐久性 ¹⁶ | | 10,000 | | | 周期 |
| 数据保持 ¹⁷ | | 20 | | | 年 |
| 数字输入 | 除NTRST外的所有数字输入 | | | | |
| 输入漏电流 | 输入(高)= DVDD | | ±1 | ±10 | μA |
| 输入上拉电流 | 输入(低)= 0 V | 10 | 20 | 80 | μA |
| 输入电容 | | | 10 | | pF |
| 输入漏电流 | 仅NTRST: 输入(低)= 0 V | | ±1 | ±10 | μA |
| 输入下拉电流 | 仅NTRST: 输入(高)= DVDD | 30 | 55 | 100 | μA |
| 逻辑输入 ¹ | 所有逻辑输入 | | | | |
| 输入低电压(VINL) | | | | 0.4 | V |
| 输入高电压(VINH) | | 2.0 | | | V |
| 逻辑输出 ¹ | 除XTALO外的所有逻辑输出 | | | | |
| 输出低电压(VOL) | I _{SOURCE} = 1.6 mA | | | 0.6 | V |
| 输出高电压(VOH) | I _{SOURCE} = 1.6 mA | 2.0 | | | V |
| 晶体振荡器 ¹ | | | | | |
| 逻辑输入, 仅限XTALI | | | | | |
| 输入低电压(VINL) | | | | 0.8 | V |
| 输入高电压(VINH) | | 1.7 | | | V |
| XTALI电容 | | | 12 | | pF |
| XTALO电容 | | | 12 | | pF |
| 片内振荡器 | | | | | |
| 振荡器 | | | 32,768 | | kHz |
| 精度 | | -3 | | +3 | % |
| MCU时钟速率 | 在此范围内八个可编程内核时钟 选择: 二进制除法1, 2, 4, 8 .. 64, 128 | 0.08 | 1.28 | 10.24 | MHz |
| 使用外部时钟至P2.0/EXTCLK引脚 | | 0.08 | | 10.24 | MHz |
| MCU启动时间 | | | | | |
| 上电时 | 包括内核上电执行时间 | | 134 | | ms |
| 复位事件后 | 包括内核上电执行时间 | | 5 | | ms |
| 从MCU关断起 | | | | | |
| PLL开启 | | | | | |
| 中断唤醒 | CD = 0 | | 4.8 | | μs |
| PLL关闭 | | | | | |
| 中断唤醒 | CD = 0 | | 66 | | μs |
| 内部PLL锁定时间 | | | 1 | | ms |

ADuC7060/ADuC7061

| 参数 | 测试条件/注释 | 最小值 | 典型值 | 最大值 | 单位 |
|---|---|-------|----------|-------|----|
| 电源要求 | | | | | |
| 电源电压 | | | | | |
| DVDD (±5%) | | 2.375 | 2.5 | 2.625 | V |
| AVDD (±5%) | | 2.375 | 2.5 | 2.625 | V |
| 功耗 | | | | | |
| I _{DD} (MCU正常模式) ¹⁸ | MCU时钟速率 = 10.24 MHz, ADC0开启 | | 6 | 10 | mA |
| | MCU时钟速率 = 640 kHz, ADC0开启, G = 4, ADC1/DAC 关闭, SPI开启; POWCON1 = 0x4 | | | 3.1 | mA |
| | 全温度范围 | | | 2.74 | mA |
| | 缩小的温度范围-40°C至+85°C ¹ | | | 350 | μA |
| I _{DD} (MCU关断) ¹ | 全温度范围 | | 55 | 120 | μA |
| | 缩小的温度范围: -40°C至+85°C | | 55 | | μA |
| I _{DD} (主ADC) | PGA使能、正常模式/低功耗模式、 电流取决于增益设置。 | | 0.6/0.3 | | mA |
| | ADC0开启, G = 1, 正常模式 | | 0.03 | | mA |
| | ADC0开启, G = 4, 正常模式 | | 0.44 | | mA |
| | ADC0开启, G = >128, 正常模式 | | 0.63 | | mA |
| I ^{DD} (辅助ADC) | 正常模式/低功耗模式 | | 0.35/0.1 | | mA |
| I _{DD} (DAC) | DAC0CON = 0x10 | | 0.33 | | mA |
| PWM | | | 0.34 | | mA |

¹ 这些参数未经过产品测试，但在产品发布时由设计和/或特性数据保证。

² 对于主ADC增益设置PGA = 4至64有效。

³ 初始失调校准后在增益范围= 4时测试。

⁴ 利用内部短路测定。系统零电平校准消除此误差。ADC工厂校准在斩波关闭情况下执行。

⁵ 利用内部短路测定。

⁶ 这些参数不包括内部基准电压温度漂移。

⁷ 增益 = 1时工厂校准。

⁸ 在具体增益范围内的系统校准可以消除在此范围内的误差。ADC工厂校准在斩波关闭情况下执行。

⁹ 采用外部基准电压测量。

¹⁰ 设置共模电压时，应确保VIN*增益设置，即PGA输出电压，在范围0.1 V至VDD - 0.7 V内。在所有增益中，900 mV是共模电压最优值。

¹¹ 利用盒子方法测定。

¹² 通过设置ADC0CON位12，可使用高达AVDD的基准电压。

¹³ 参考DAC的线性度是使用一个递减的数据范围171到4095计算出来的。

¹⁴ 参考DAC的线性度是使用一个递减的数据范围2731到65,535计算出来的。

¹⁵ 芯片温度。

¹⁶ 耐久性是在-40°C、+25°C及+125°C时依据JEDEC 22标准方法A117认定为10,000个周期。在25°C时的典型耐久性为170,000个周期。

¹⁷ 根据JEDEC22标准方法A117使用寿命相当于+ 85°C结温时的寿命。保持期限会随着结温递减。

¹⁸ 在Flash/EE存储器编程和擦除周期期间的典型额外电源电流消耗分别为7 mA和5 mA。

时序规格

I²C时序

表2. 标准模式下I²C®时序(100 kHz)

| 参数 | 描述 | 从机 | | 单位 |
|-----------|----------------------|-----|------|---------------|
| | | 最小值 | 最大值 | |
| t_L | SCLOCK低电平脉冲宽度 | 4.7 | | μs |
| t_H | SCLOCK高电平脉冲宽度 | 4.0 | | ns |
| t_{SHD} | 起始条件保持时间 | 4.0 | | μs |
| t_{DSU} | 数据建立时间 | 250 | | ns |
| t_{DHD} | 数据保持时间 | 0 | 3.45 | μs |
| t_{RSU} | 重复起始建立时间 | 4.7 | | μs |
| t_{PSU} | 结束条件建立时间 | 4.0 | | μs |
| t_{BUF} | 一个结束条件和起始条件之间的总线空闲时间 | 4.7 | | μs |
| t_R | CLOCK和SDATA上升时间 | | 1 | μs |
| t_F | CLOCK和SDATA下降时间 | | 300 | ns |

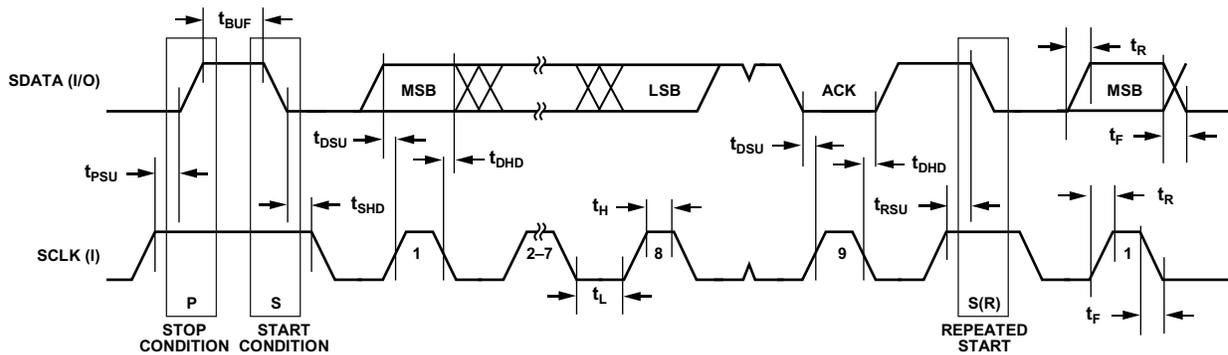


图2. I²C兼容接口时序

07079-02

ADuC7060/ADuC7061

SPI时序

表3. SPI主机定时(相位模式 = 1)

| 参数 | 描述 | 最小值 | 典型值 | 最大值 | 单位 |
|-----------|---------------------------------|---------------------|--------------------------------|-----|----|
| t_{SL} | SCLOCK低电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{SH} | SCLOCK高电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{DAV} | SCLOCK边沿之后数据输出有效 | | | 25 | ns |
| t_{DSU} | SCLOCK边沿之前数据输入建立时间 ¹ | $1 \times t_{UCLK}$ | | | ns |
| t_{DHD} | SCLOCK边沿之后数据输入保持时间 ¹ | $2 \times t_{UCLK}$ | | | ns |
| t_{DF} | 数据输出下降时间 | | 30 | 40 | ns |
| t_{DR} | 数据输出上升时间 | | 30 | 40 | ns |
| t_{SR} | SCLOCK上升时间 | | 30 | 40 | ns |
| t_{SF} | SCLOCK下降时间 | | 30 | 40 | ns |

¹ $t_{UCLK} = 97.6$ ns。它对应于由PLL产生的10.24 MHz内部时钟。

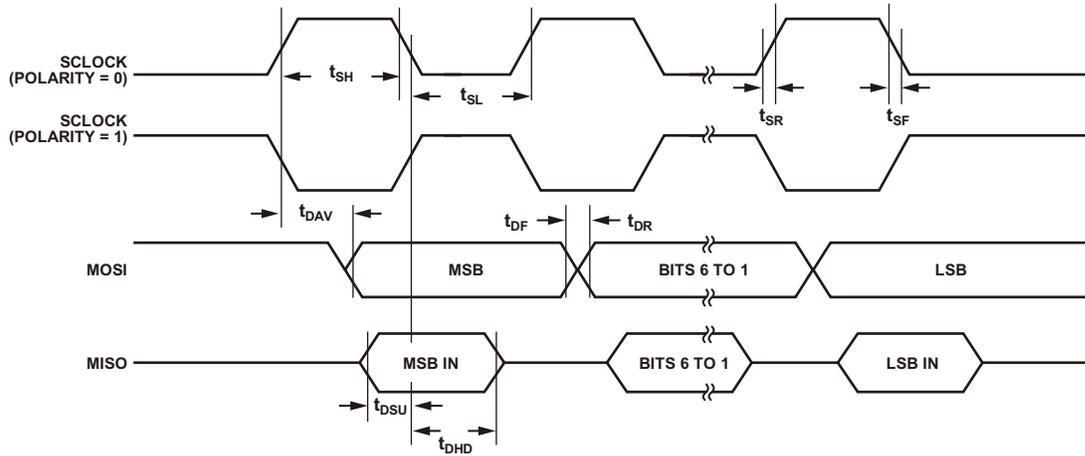


图3. SPI主机定时(相位模式 = 1)

表4. SPI主机定时(相位模式 = 0)

| 参数 | 描述 | 最小值 | 典型值 | 最大值 | 单位 |
|------------|---------------------------------|---------------------|--------------------------------|-----|----|
| t_{SL} | SCLOCK低电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{SH} | SCLOCK高电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{DAV} | SCLOCK边沿之后数据输出有效 | | | 25 | ns |
| t_{DOSU} | SCLOCK边沿之前数据输出建立 | | | 90 | ns |
| t_{DSU} | SCLOCK边沿之前数据输入建立时间 ¹ | $1 \times t_{UCLK}$ | | | ns |
| t_{DHD} | SCLOCK边沿之后数据输入保持时间 ¹ | $2 \times t_{UCLK}$ | | | ns |
| t_{DF} | 数据输出下降时间 | | 30 | 40 | ns |
| t_{DR} | 数据输出上升时间 | | 30 | 40 | ns |
| t_{SR} | SCLOCK上升时间 | | 30 | 40 | ns |
| t_{SF} | SCLOCK下降时间 | | 30 | 40 | ns |

¹ $t_{UCLK} = 97.6$ ns。它对应于由PLL产生的10.24 MHz内部时钟。

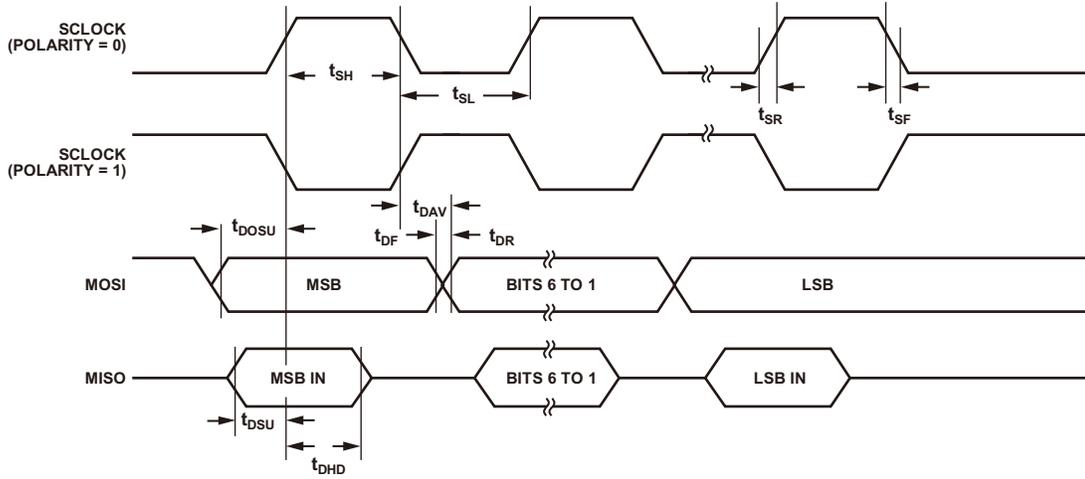


图4. SPI主机定时(相位模式 = 0)

07079-031

表5. SPI从机定时(相位模式 = 1)

| 参数 | 描述 | 最小值 | 典型值 | 最大值 | 单位 |
|---------------------|--|---|--------------------------------|-----|----|
| $t_{\overline{CS}}$ | \overline{CS} 到SCLOCK边沿 ¹ | $(2 \times t_{HCLK}) + (2 \times t_{UCLK})$ | | | ns |
| t_{SL} | SCLOCK低电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{SH} | SCLOCK高电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{DAV} | SCLOCK边沿之后数据输出有效 | | | 40 | ns |
| t_{DSU} | SCLOCK边沿之前数据输入建立时间 ¹ | $1 \times t_{UCLK}$ | | | ns |
| t_{DHD} | SCLOCK边沿之后数据输入保持时间 ¹ | $2 \times t_{UCLK}$ | | | ns |
| t_{DF} | 数据输出下降时间 | | 30 | 40 | ns |
| t_{DR} | 数据输出上升时间 | | 30 | 40 | ns |
| t_{SR} | SCLOCK上升时间 | 1 | | | ns |
| t_{SF} | SCLOCK下降时间 | 1 | | | ns |
| t_{SFS} | \overline{CS} 在SCLOCK边沿后变高 | 0 | | | ns |

¹ $t_{UCLK} = 97.6$ ns。它对应于由PLL产生的10.24 MHz内部时钟。

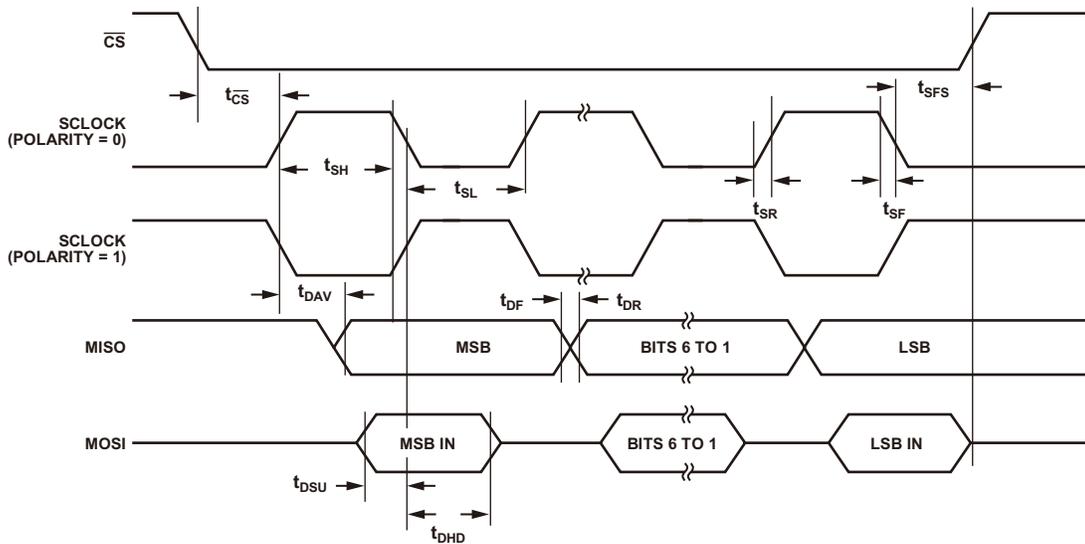


图5. SPI从机定时(相位模式 = 1)

07079-032

ADuC7060/ADuC7061

表6. SPI从机定时(相位模式 = 0)

| 参数 | 描述 | 最小值 | 典型值 | 最大值 | 单位 |
|---------------------|--|---|--------------------------------|-----|----|
| $t_{\overline{CS}}$ | \overline{CS} 到SCLOCK边沿 ¹ | $(2 \times t_{HCLK}) + (2 \times t_{UCLK})$ | | | ns |
| t_{SL} | SCLOCK低电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{SH} | SCLOCK高电平脉冲宽度 | | $(SPIDIV + 1) \times t_{HCLK}$ | | ns |
| t_{DAV} | SCLOCK边沿之后数据输出有效 | | | 40 | ns |
| t_{DSU} | SCLOCK边沿之前数据输入建立时间 ¹ | $1 \times t_{UCLK}$ | | | ns |
| t_{DHD} | SCLOCK边沿之后数据输入保持时间 ¹ | $2 \times t_{UCLK}$ | | | ns |
| t_{DF} | 数据输出下降时间 | | 30 | 40 | ns |
| t_{DR} | 数据输出上升时间 | | 30 | 40 | ns |
| t_{SR} | SCLOCK上升时间 | 1 | | | ns |
| t_{SF} | SCLOCK下降时间 | 1 | | | ns |
| t_{DOCS} | \overline{CS} 边沿之后数据输出有效 | | | 10 | ns |
| t_{SFS} | \overline{CS} 在SCLOCK边沿后变高 | 0 | | | ns |

¹ $t_{UCLK} = 97.6$ ns。它对应于由PLL产生的10.24 MHz内部时钟。

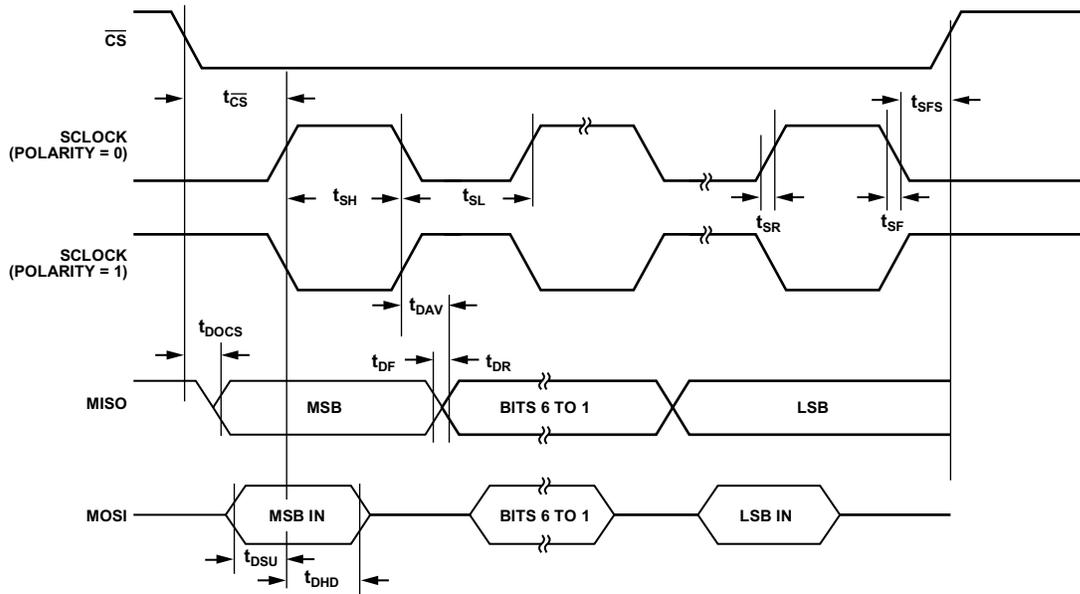


图6. SPI从机定时(相位模式 = 0)

07079-033

绝对最大额定值

$T_A = -40^{\circ}\text{C}$ 至 $+125^{\circ}\text{C}$ (除非另有说明)。

表7.

| 参数 | 额定值 |
|---------------------|---------------------|
| AGND至DGND至AVDD至DVDD | -0.3 V至+0.3 V |
| 数字I/O电压至DGND | -0.3 V至+3.6 V |
| VREF±至AGND | -0.3 V至AVDD + 0.3 V |
| ADC输入至AGND | -0.3 V至AVDD + 0.3 V |
| ESD(人体模型)额定值 | |
| 所有引脚 | ±2 kV |
| 存储温度 | 125°C |
| 结温 | |
| 瞬时 | 150°C |
| 连续 | 130°C |
| 引脚温度 | |
| 回流焊(15秒) | 260°C |

注意，超出上述绝对最大额定值可能会导致器件永久性损坏。这只是额定应力值，不涉及器件在这些或任何其它条件下超出本技术规格指标的功能性操作。长期在绝对最大额定值条件下工作会影响器件的可靠性。

ESD警告

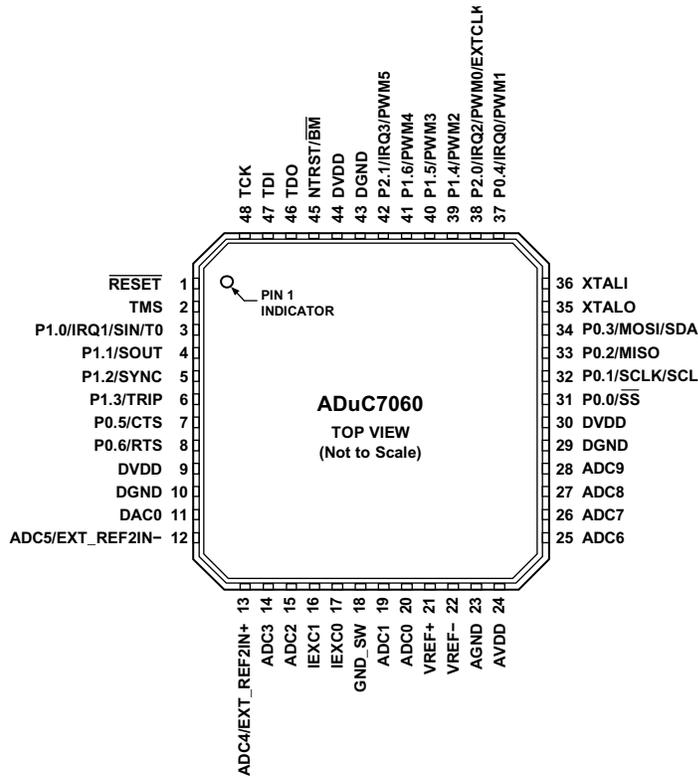


ESD(静电放电)敏感器件。

带电器件和电路板可能会在没有察觉的情况下放电。尽管本产品具有专利或专有保护电路，但在遇到高能量ESD时，器件可能会损坏。因此，应当采取适当的ESD防范措施，以避免器件性能下降或功能丧失。

ADuC7060/ADuC7061

引脚配置和功能描述



NOTES
1. THE LFCSP_VQ ONLY HAS AN EXPOSED PADDLE THAT MUST BE LEFT UNCONNECTED. THIS DOES NOT APPLY TO THE LQFP.

07079-002

图7. 48引脚LQFP和48引脚LFCSP_VQ封装的引脚配置

表8. ADuC7060引脚功能描述

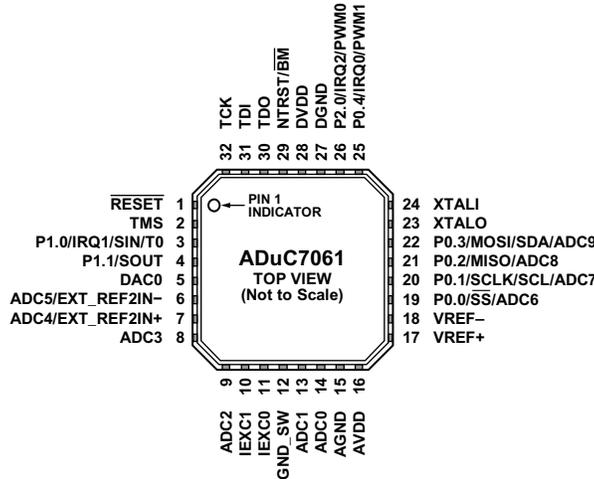
| 引脚编号 | 名称 | 类型 ¹ | 描述 |
|------|------------------|-----------------|---|
| 0 | EP | | 底部焊盘。可以接地或悬空。LQFP封装没有底部焊盘。 |
| 1 | RESET | I | 复位。输入引脚(低电平有效)。建议将该引脚与一个1 kΩ的外部上拉电阻相连。 |
| 2 | TMS | I | JTAG测试模式选择。输入引脚，用于调试或者下载。应在该引脚处增加一个约100 kΩ的外部上拉电阻。 |
| 3 | P1.0/IRQ1/SIN/T0 | I/O | 通用输入输出P1.0/外部中断请求1/串行输入/定时器0输入。这是一个多功能输入输出引脚，可提供四种功能。 |
| 4 | P1.1/SOUT | I/O | 通用输入输出P1.1/串行输出。这是一个双功能输入/输出引脚。 |
| 5 | P1.2/SYNC | I/O | 通用输入输出P1.2/PWM外部同步输入。这是一个双功能输入/输出引脚。 |
| 6 | P1.3/TRIP | I/O | 通用输入输出P1.3/PWM外部触发输入。这是一个双功能输入/输出引脚。 |
| 7 | P0.5/CTS | I/O | 通用输入输出P0.5/UART模式下清除发送信号。 |
| 8 | P0.6/RTS | I/O | 通用输入输出P0.6/UART模式下清除发送信号。 |
| 9 | DVDD | S | 数字电源引脚。 |
| 10 | DGND | S | 数字地。 |
| 11 | DAC0 | O | DAC输出。模拟输出引脚。 |

| 引脚编号 | 名称 | 类型 ¹ | 描述 |
|------|-----------------------|-----------------|---|
| 12 | ADC5/EXT_REF2IN- | I | 单端或差分模拟输入5/外部基准负输入端.这是一个双功能模拟输入引脚。ADC5为辅助ADC提供模拟输入。EXT_REF2IN-可为辅助通道提供ADC外部基准负输入。 |
| 13 | ADC4/EXT_REF2IN+ | I | 多功能模拟输入引脚。该引脚可作为单端或差分模拟输入4, 为辅助ADC提供模拟输入; 另外, 它还可以为辅助通道提供外部基准正输入电压。 |
| 14 | ADC3 | I | 单端或差分模拟输入3.为主ADC和辅助ADC提供模拟输入。 |
| 15 | ADC2 | I | 单端或差分模拟输入2.为主ADC和辅助ADC提供模拟输入。 |
| 16 | IEXC1 | O | 可编程电流源模拟输出引脚。 |
| 17 | IEXC0 | O | 可编程电流源模拟输出引脚。 |
| 18 | GND_SW | I | 切换至内部模拟地基准。不用时, 直接接到AGND系统地。 |
| 19 | ADC1 | I | 单端或差分模拟输入1.为主ADC提供模拟输入。为主ADC提供负差分输入。 |
| 20 | ADC0 | I | 单端或差分模拟输入0.为主ADC提供模拟输入。为主ADC提供正差分输入。 |
| 21 | VREF+ | I | 为主通道提供外部基准正输入。模拟输入引脚。 |
| 22 | VREF- | I | 为主通道提供外部基准负输入。模拟输入引脚。 |
| 23 | AGND | S | 模拟地。 |
| 24 | AVDD | S | 模拟电源引脚。 |
| 25 | ADC6 | I | 模拟输入6, 为辅助ADC提供模拟输入。单端或差分模拟输入6。 |
| 26 | ADC7 | I | 模拟输入7, 为辅助ADC提供模拟输入。单端或差分模拟输入7。 |
| 27 | ADC8 | I | 模拟输入8, 为辅助ADC提供模拟输入。单端或差分模拟输入8。 |
| 28 | ADC9 | I | 模拟输入9, 为辅助ADC提供模拟输入。单端或差分模拟输入9。 |
| 29 | DGND | S | 数字地。 |
| 30 | DVDD | S | 数字电源引脚。 |
| 31 | P0.0/ \overline{SS} | I/O | 通用输入输出P0.0/SPI从机选择引脚(低电平有效)。这是一个双功能输入/输出引脚。 |
| 32 | P0.1/SCLK/SCL | I/O | 通用输入输出P0.1/SPI时钟引脚/I ² C时钟引脚。这是一个三功能输入/输出引脚。 |
| 33 | P0.2/MISO | I/O | 通用输入输出P0.2/SPI主机输入从机输出。这是一个双功能输入/输出引脚。 |
| 34 | P0.3/MOSI/SDA | I/O | 通用输入输出P0.3/SPI主机输出从机输入/I ² C数据引脚。这是一个三功能输入/输出引脚。 |
| 35 | XTALO | O | 外部晶体振荡器输出引脚。 |
| 36 | XTALI | I | 外部晶体振荡器输入引脚。 |
| 37 | P0.4/IRQ0/PWM1 | I/O | 通用输入输出P0.4/外部中断请求0/PWM1输出。这是一个三功能输入/输出引脚。 |
| 38 | P2.0/IRQ2/PWM0/EXTCLK | I/O | 通用输入输出P2.0/外部中断请求2/PWM0输出/外部时钟输入。这是一个多功能输入/输出引脚。 |
| 39 | P1.4/PWM2 | I/O | 通用输入输出P1.4/PWM2输出。这是一个双功能输入/输出引脚。 |
| 40 | P1.5/PWM3 | I/O | 通用输入输出P1.5/PWM3输出。这是一个双功能输入/输出引脚。 |
| 41 | P1.6/PWM4 | I/O | 通用输入输出P1.6/PWM4输出。这是一个双功能输入/输出引脚。 |
| 42 | P2.1/IRQ3/PWM5 | I/O | 通用输入输出P2.1/外部中断请求3/PWM5输出。这是一个三功能输入/输出引脚。 |

ADuC7060/ADuC7061

| 引脚编号 | 名称 | 类型 ¹ | 描述 |
|------|-------------------------------|-----------------|---|
| 43 | DGND | S | 数字地。 |
| 44 | DVDD | S | 数字电源引脚。 |
| 45 | NTRST/ $\overline{\text{BM}}$ | I | JTAG复位/启动模式。输入引脚，用于调试、下载和启动模式($\overline{\text{BM}}$)。当复位时，如果 $\overline{\text{BM}}$ 为低电平，ADuC7060将进入串行下载模式；在复位时，如果通过13 k Ω 的电阻将 $\overline{\text{BM}}$ 拉高ADuC7060将执行代码。 |
| 46 | TDO | O | JTAG数据输出。输出引脚，用于调试或者下载。 |
| 47 | TDI | I | JTAG数据输入。输入引脚，用于调试或者下载。应在该引脚处增加一个约100 k Ω 的外部上拉电阻。 |
| 48 | TCK | I | JTAG时钟输入。输入引脚，用于调试或者下载。应在该引脚处增加一个约100 k Ω 的外部上拉电阻。 |

¹ I = 输入，O = 输出，I/O = 输入/输出，S = 电源。



NOTES
 1. THE 32-LEAD LFCSP_VQ HAS AN EXPOSED PADDLE. THIS EXPOSED PADDLE MUST BE LEFT UNCONNECTED.

07079-903

图8. 32引脚LFCSP的引脚配置

表9. ADuC7061引脚功能描述

| 引脚编号 | 名称 | 类型 ¹ | 描述 |
|------|--------------------|-----------------|--|
| 0 | EP | | 底部焊盘。可以接地或悬空。 |
| 1 | RESET | I | 复位引脚。输入引脚(低电平有效)。建议将该引脚与一个1 kΩ的外部上拉电阻相连。 |
| 2 | TMS | I | JTAG测试模式选择。输入引脚, 用于调试或者下载。应在该引脚处增加一个约100 kΩ的外部上拉电阻。 |
| 3 | P1.0/IRQ1/SIN/T0 | I/O | 通用输入输出P1.0/外部中断请求1/串行输入/定时器0输入。这是一个多功能输入输出引脚, 可提供四种功能。 |
| 4 | P1.1/SOUT | I/O | 通用输入输出P1.1/串行输出。这是一个双功能输入/输出引脚。 |
| 5 | DAC0 | O | DAC输出。模拟输出引脚。 |
| 6 | ADC5/EXT_REF2IN- | I | 单端或差分模拟输入5/外部基准负输入端。这是一个双功能模拟输入引脚。ADC5为辅助ADC提供模拟输入。EXT_REF2IN-可为辅助通道提供ADC外部基准负输入。 |
| 7 | ADC4/EXT_REF2IN+ | I | 多功能模拟输入引脚。该引脚可作为单端或差分模拟输入端, 为辅助ADC提供模拟输入4; 另外, 它还可以为辅助通道提供外部基准正输入电压。 |
| 8 | ADC3 | I | 单端或差分模拟输入3。为主ADC和辅助ADC提供模拟输入。 |
| 9 | ADC2 | I | 单端或差分模拟输入2。为主ADC和辅助ADC提供模拟输入。 |
| 10 | IEXC1 | O | 可编程电流源模拟输出引脚。 |
| 11 | IEXC0 | O | 可编程电流源模拟输出引脚。 |
| 12 | GND_SW | I | 切换至内部模拟地基准。不用时, 直接接到AGND系统地。 |
| 13 | ADC1 | I | 单端或差分模拟输入1。为主ADC提供模拟输入。为主ADC提供负差分输入。 |
| 14 | ADC0 | I | 单端或差分模拟输入0。为主ADC提供模拟输入。为主ADC提供正差分输入。 |
| 15 | AGND | S | 模拟地。 |
| 16 | AVDD | S | 模拟电源引脚。 |
| 17 | VREF+ | I | 为主通道提供外部基准正输入。模拟输入引脚。 |
| 18 | VREF- | I | 为主通道提供外部基准负输入。模拟输入引脚。 |
| 19 | P0.0/SS/ADC6 | I/O | 通用输入输出P0.0/SPI从机选择(低电平有效)/辅助ADC6输入引脚。这是一个多功能输入/输出引脚。单端或差分模拟输入6。为辅助ADC提供模拟输入。 |
| 20 | P0.1/SCLK/SCL/ADC7 | I/O | 通用输入输出P0.1/SPI时钟/I ² C时钟/辅助ADC7输入引脚。这是一个多功能输入/输出引脚。单端或差分模拟输入7。为辅助ADC提供模拟输入。 |

ADuC7060/ADuC7061

| 引脚编号 | 名称 | 类型 ¹ | 描述 |
|------|-------------------------------|-----------------|--|
| 21 | P0.2/MISO/ADC8 | I/O | 通用输入输出P0.2/SPI主机输入从机输出/辅助ADC8输入。这是一个三功能输入/输出引脚。单端或差分模拟输入8。为辅助ADC提供模拟输入。 |
| 22 | P0.3/MOSI/SDA/ADC9 | I/O | 通用输入输出P0.3/SPI主机输出从机输入/I ² C数据引脚/辅助ADC9输入。这是一个多功能输入/输出引脚。单端或差分模拟输入9。为辅助ADC提供模拟输入。 |
| 23 | XTALO | O | 外部晶体振荡器输出引脚。 |
| 24 | XTALI | I | 外部晶体振荡器输入引脚。 |
| 25 | P0.4/IRQ0/PWM1 | I/O | 通用输入输出P0.4/外部中断请求0/PWM1输出。这是一个三功能输入/输出引脚。 |
| 26 | P2.0/IRQ2/PWM0 | I/O | 通用输入输出P2.0/外部中断请求2/PWM0输出。这是一个三功能输入/输出引脚。 |
| 27 | DGND | S | 数字地。 |
| 28 | DVDD | S | 数字电源引脚。 |
| 29 | NTRST/ $\overline{\text{BM}}$ | I | JTAG复位/启动模式。输入引脚，用于调试、下载和启动模式($\overline{\text{BM}}$)。当复位时，如果 $\overline{\text{BM}}$ 为低电平，ADuC7061将进入串行下载模式；在复位时，如果通过13 k Ω 的电阻将 $\overline{\text{BM}}$ 拉高，ADuC7061将执行代码。 |
| 30 | TDO | O | JTAG数据输出。输出引脚，用于调试或者下载。 |
| 31 | TDI | I | JTAG数据输入。输入引脚，用于调试或者下载。应在该引脚处增加一个约100 k Ω 的外部上拉电阻。 |
| 32 | TCK | I | JTAG时钟。输入引脚，用于调试或者下载。应在该引脚处增加一个约100 k Ω 的外部上拉电阻。 |

¹ I = 输入，O = 输出，I/O = 输入/输出，S = 电源。

术语

转换速率

转换速率指在ADC稳定后，从ADC获得输出结果的速度。

此器件采用了 Σ - Δ 转换技术，这意味着当以相对较高的采样率对ADC前端信号进行过采样时，后接数字滤波器将抽取输出数据，以1 Hz至8 kHz的输出速率输出一个24位有效数据转换结果。

注意，当软件在同一ADC上切换不同输入时，必须先清空数字滤波器，然后对新结果求平均值。这个操作可能需要花费多个转换周期，具体的周期数取决于ADC的配置和滤波器的类型。

积分非线性(INL)

转换结果编码偏离通过其传递函数端点的直线的最大偏差。传递函数端点是指，在零点位置比第一个编码的跃变点低 $\frac{1}{2}$ LSB的点，以及在满刻度位置比最后一个编码的跃变点高 $\frac{1}{2}$ LSB的点。(111 ... 110至111... 111)该误差表示为满量程的百分比。

无失码

无失码是衡量ADC微分非线性的指标。误差用位来表示(2N位，其中N为无失码)，指在整个ADC输入范围内保证发生的代码数目(ADC输出结果)。

失调误差

ADC第一个码转换输入电压和理想码跃迁的偏差即是失调误差。

失调误差漂移

失调误差漂移是指绝对失调误差的温度变化值。该误差表示为最低有效位/摄氏度，即LSB/ $^{\circ}$ C。

增益误差

增益误差是衡量ADC量程误差的指标。它考察的是传递函

输出噪声

输出噪声指在ADC直流信号输入后，所产生的输出代码分布的标准差(或 1σ)。它用微均方根来表示。输出噪声，即均方根噪声，可用来计算ADC的有效分辨率；计算公式为：

$$\text{有效分辨率} = \log_2(\text{满量程}/\text{均方根噪声})\text{位}$$

峰峰值噪声指ADC直流信号输入时在输出代码分布 $6.6 \times \Sigma$ 范围内的代码偏差。因此，峰峰值噪声计算公式如下：

$$6.6 \times \text{均方根噪声}$$

峰峰值噪声可用来计算在 $6.6 \times \Sigma$ 限值内无码闪烁的ADC(无噪声代码)分辨率；计算公式如下：

$$\text{无噪声代码分辨率} = \log_2 \text{位}$$

数据手册中用到的首字母缩写词

| | |
|------|-----------|
| ADC | 模数转换器 |
| ARM | 高级精简指令集机器 |
| JTAG | 联合测试行动小组 |
| LSB | 最低有效字节/位 |
| LVF | 低电压标志 |
| MCU | 微控制器 |
| 寄存器 | 存储器映射寄存器 |
| MSB | 最高有效字节/位 |
| PID | 保护标识符 |
| POR | 上电复位 |
| PSM | 电源监控器 |
| rms | 均方根 |

ARM7DMI内核概览

ARM7®内核是一款由ARM®公司开发的32位精简指令集计算机(RISC)。ARM7TDMI采用冯诺依曼架构，指令和数据使用单32位总线。它支持8位、16位和32位数据长度，指令字长度可为16位或32位，具体的位数取决于内核的工作模式。

ARM7TDMI采用ARM7内核，具有四种附加特性，如表10所列。

表10. ARM7TDMI特性

| 特性 | 描述 |
|----|-----------------------------|
| T | 支持Thumb®指令集(16位) |
| D | 支持调试 |
| M | 增强乘法器 |
| I | 包含EmbeddedICE™模块，以支持嵌入式系统调试 |

THUMB模式(T)

一条ARM指令的长度为32位。ARM7TDMI处理器支持第二个指令集，该指令集被压缩成16位，称为thumb指令集。采用Thumb指令集，可以拥有更快的代码执行速度(16位存储器)和更高的代码密度，因此，ARM7TDMI内核特别适用于嵌入式应用。

然而，Thumb模式有三个缺点：

- 与ARM相比，在执行同一个任务时，Thumb代码通常需要更多的指令。因此，ARM代码特别适合于在大多数应用中最大程度提升时间关键代码的性能。
- Thumb指令集不含一些异常处理指令，所以可能需要ARM代码来进行异常处理。
- 发生中断时，内核矢量指向存储器内的中断地址并执行该地址单元内的代码。第一条命令须包含在ARM代码内。

乘法器(M)

ARM7TDMI指令集包括一个增强乘法器，具有四条额外指令用来进行32位与32位相乘或32位与32位乘加(MAC)，得到64位结果。

EmbeddedICE (I)

EmbeddedICE模块为ARM7TDMI提供集成片内调试功能。它包括断点和观察点寄存器，可用来实现非介入用户代码调试。这些寄存器可以通过JTAG测试端口来控制。当遇到一个断点或观察点时，处理器会暂停运行，进入调试状态。当处理器处于调试状态，就可以查询处理器寄存器，也可以查询Flash/EE、SRAM和存储器映射寄存器。

ARM7异常

ARM7支持5种类型的异常，每种类型分别对应一种优先处理模式。这5种异常类型为：

第一种类型：正常中断或IRQ。这种异常用于内、外部事件的通用中断处理。注意，ADuC7060/ADuC7061为所有IRQ源提供8个可配置的优先级。

第二种类型：快速中断或FIQ。这种异常用于数据传输或低延迟通信通道处理。FIQ的优先级高于IRQ。注意，ADuC7060/ADuC7061为所有FIQ源提供8个可配置的优先级。

第三种类型：存储器中止(预取和数据)。

第四种类型：企图执行未定义指令。

第五种类型：软件中断(SWI)指令，用于调用操作系统。

一般情况下，程序员将中断定义为IRQ，但要获得优先级更高的中断，可将中断定义为FIQ类型。

异常优先级及其矢量地址见表11。

表11.异常优先级和矢量地址

| 优先级 | 异常 | 异常 |
|-----|-----------|------|
| 1 | 硬件复位 | 0x00 |
| 2 | 存储器中止(数据) | 0x10 |
| 3 | FIQ | 0x1C |
| 4 | IRQ | 0x18 |
| 5 | 存储器中止(预取) | 0x0C |
| 6 | 软件中断1 | 0x08 |
| 6 | 未定义指令1 | 0x04 |

¹ 软件中断和未定义指令异常的优先级相同且互斥。

在表11中所列异常的地址范围为0x00 to 0x1C，其中，0x14为保留地址。

ARM寄存器

ARM7TDMI共有16个标准寄存器。R0至R12用于数据操作；R13是堆栈指针；R14是链接寄存器；而R15是程序计数器，用来指示当前正在执行的指令。链接寄存器用来保存用户转移地址(在使用转移和链接命令的情况下)或发生异常时的命令。

堆栈指针用于保存堆栈的当前位置。通常，在ARM7TDMI内，堆栈的起始地址为可用RAM空间顶部，并根据需要递减。每个异常分别对应一个堆栈。用户可根据目标应用的需要来配置每个堆栈的大小。利用C语言等高级语言进行编程时，必须确保不产生堆栈溢出。这取决于所用编译器的性能。

ADuC7060/ADuC7061

重映射

ARM异常矢量位于存储器阵列底部，其存储地址范围为0x00000000至0x00000020。

在默认情况下，在复位后，Flash/EE存储器会被逻辑映射到地址0x00000000。通过将位于0xFFFF0220地址的重映射寄存器的第0位置位，可以将SRAM逻辑重映射到地址0x00000000。将重映射寄存器的第0位清零，可让Flash/EE返回到0x00000000。

将RAM重映射到0x00000000，有利于优化ADuC7060/ADuC7061的中断延迟时间；因为在这种情况下，代码可以在完全32位ARM模式下以最高内核速度执行。注意，发生异常时，内核默认为ARM模式。

重映射操作

ADuC7060/ADuC7061发生复位时，会自动执行工厂内置的程序代码。这就是所谓的隐藏内核，用户代码不能访问。如果ADuC7060/ADuC7061处于正常模式，它首先执行内核的上电配置例行程序，然后跳转到复位矢量地址0x00000000，执行用户的复位异常例行程序。由于复位时Flash/EE镜像到存储器阵列底部，复位例行程序必须始终写入Flash/EE内。

重映射命令必须从绝对Flash/EE地址处开始执行，而不能被镜像、存储器重映射段处开始执行，因为该段可能被SRAM所取代。如果在从镜像位置开始执行代码时执行重映射操作，可能会发生预取/数据中止或用户可能观察到异常程序操作。任何一种复位都会将Flash/EE存储器逻辑重映射到存储器阵列底部。

REMAP寄存器

名称：重映射
地址：0xFFFF0220
默认值：0x0000
访问类型：读、写
功能：该8位寄存器允许用户代码将RAM或Flash/EE存储空间重映射到ARM存储器空间底部(起始地址为0x00000000)。

表12.REMAP寄存器位功能描述

| 位 | 描述 |
|-----|--|
| 7:1 | 保留。这些位保留，应由用户代码写入0。 |
| 0 | 重映射位。 用户置1，可将SRAM映射到0x00000000地址处。 重映射Flash/EE到地址0x00000000后，该位自动清0。 |

Flash/EE控制接口

Flash/EE控制接口用于串行下载和JTAG编程，包括本节所列的8个寄存器。注意，每个Flash页的容量为512个字节。

FEESTA寄存器

FEESTA是一个只读寄存器，它反映Flash/EE控制接口的状态。如表13所示。

名称：FEESTA
地址：0xFFFF0E00
默认值：0x0020
访问类型：读

表13.FEESTA寄存器位功能描述

| 位 | 描述 |
|------|---|
| 15:6 | 保留。 |
| 5 | 保留。 |
| 4 | 保留。 |
| 3 | Flash中断状态位。当中断发生后，即当一条命令执行完毕且FEEMOD寄存器中的Flash/EE中断使能位被置1，Flash中断状态位自动置1。读FEESTA寄存器后，该位清0。 |
| 2 | Flash/EE控制器忙。当控制器忙时，该位自动置1；当控制器空闲时，该位自动清0。 |
| 1 | 命令失败。当命令的执行失败时，该位自动置1；读FEESTA寄存器后，该位清0。 |
| 0 | 指令通过。当命令成功执行时，MicroConverter®将该位由置1；读FEESTA寄存器后，该位清0。 |

FEEMOD寄存器

FEEMOD用来设置Flash/EE控制接口的工作模式。表14显示FEEMOD寄存器的位分配。

名称：FEEMOD
地址：0xFFFF0E00
默认值：0x0020
访问类型：读

表14.FEEMOD 寄存器位功能描述

| 位 | 描述 |
|------|--|
| 15:9 | 保留。 |
| 8 | 保留。始终将该位置为1。 |
| 7:5 | 保留。这些位的值应为0(写密钥的情况除外)。 |
| 4 | Flash/EE中断使能位。 用户置1, 使能Flash/EE中断。当处理器执行完一条命令后, 将产生中断。 用户清0, 禁用Flash/EE中断。 |
| 3 | 擦/写命令保护。 用户置1, 使能擦/写命令; 该位清0, 防止对Flash/EE执行擦/写命令。 |
| 2:0 | 保留。始终将这些位置为0。 |

FEECON寄存器

FEECON是一个8位命令寄存器。各命令如表15所示。

| | |
|-------|------------|
| 名称: | FEECON |
| 地址: | 0xFFFF0E08 |
| 默认值: | 0x07 |
| 访问类型: | 读 |

表15.FEECON内的命令代码

| 代码 | 命令 | 描述 |
|-------------------|------|---|
| 0x00 ¹ | 空 | 空闲状态。 |
| 0x01 ¹ | 单次读取 | 向FEEDAT内载入16位数据。由FEEADR索引。 |
| 0x02 ¹ | 单次写入 | 在FEEADR指向的地址内写入FEEDAT。操作耗时50 μs。 |
| 0x03 ¹ | 擦/写 | 擦除由FEEADR索引的存储页, 并且向由FEEADR所指向的地址内写入FEEDAT。此项操作大约需要24 ms。 |
| 0x04 ¹ | 单次验证 | 将FEEADR所指地址中的数据与FEEDAT中的数据进行比较。比较的结果存入FEESTA的位0、位1。 |
| 0x05 ¹ | 单次擦除 | 擦除由FEEADR索引的页面。 |
| 0x06 ¹ | 批量擦除 | 擦除30 KB用户空间。而2 KB的内核空间是被保护的。为防止意外操作, 执行该指令需要一个命令序列。请参阅关于“用于执行批量擦除操作的命令序列”部分。 |
| 0x07 | 保留 | 保留。 |
| 0x08 | 保留 | 保留。 |
| 0x09 | 保留 | 保留。 |
| 0x0A | 保留 | 保留。 |
| 0x0B | 签名 | 处理器运行该命令后, 会产生一个基于LFSR(线性反馈移位寄存器)的24位签名, 该签名将被写入FEESIGN寄存器。此项操作需要16,389个时钟周期。 |
| 0x0C | 保护 | 该命令只能执行一次。FEEPRO的值会被保留; 只有通过执行批量擦除命令(0x06)或者密钥才能清除该值。 |
| 0x0D | 保留 | 保留。 |
| 0x0E | 保留 | 保留。 |
| 0x0F | Ping | 无操作; 中断产生。 |

¹ 当任一上述命令执行完毕, FEECON寄存器立即读取0x07。

ADuC7060/ADuC7061

FEEDAT寄存器

FEEDAT是一个16位数据寄存器。该寄存器用来保存Flash读写命令执行的结果(数据值)。

名称: FEEDAT
地址: 0xFFFF0E0C
默认值: 0XXXXX
访问类型: 读、写

FEEADR寄存器

FEEADR是一个16位地址寄存器, 可用来存取32 kB Flash模块的各个页。用户可以操作的有效范围为: 0x0000至0x77FF。这表示一个30 kB Flash用户存储器空间。对上述地址范围外的地址单元进行读/写操作, 将会导致数据中止异常。

名称: FEEADR
地址: 0xFFFF0E10
默认值: 0x0000
访问类型: 读、写

FEESIG寄存器

FEESIG寄存器是一个24位的寄存器。当处理器执行完一条签名命令后, 该寄存器的值将被更新为一个24位签名。这个值是由签名命令控制的线性反馈移位寄存器操作的结果。

名称: FEESIG
地址: 0xFFFF0E18
默认值: 0FFFFFFF
访问类型: 读

FEEPRO寄存器

FEEPRO 寄存器在寄存器复位后提供保护。它需要一个软件密钥(见表16)。

名称: FEEPRO
地址: 0xFFFF0E1C
默认值: 0x00000000
访问类型: 读、写

FEEHID寄存器

FEEHID寄存器提供即时保护。它不需要任何软件密钥。注意, 复位后, FEEHIDE中的保护设置将清0(见表16)。

名称: FEEHID
地址: 0xFFFF0E20
默认值: 0xFFFFFFFF
访问类型: 读、写

表16. FEEPRO和FEEHID 寄存器位功能描述

| 位 | 描述 |
|------|---|
| 31 | 读保护位。 用户清零, 以保护所有代码。如将该位清零, 可防止代码从JTAG接口被读出。用户置1, 可以通过JTAG接口读取所有代码。 |
| 30 | 保护第59页(0x00087600至0x000877FF)。用户置1, 将允许对第59页进行写操作。清0, 将保护第59页。 |
| 29 | 保护第58页(0x00087400至0x000875FF)。用户置1, 将允许对第58页进行写操作。清0, 将保护第58页。 |
| 28:0 | 提供对第0~57页的写保护。每一位对应两个存储页。每个存储页的容量为512个字节。位0用于保护第0页和1页(0x00080000至0x000803FF)。用户置1, 将允许对第0页和1页进行写操作。清0, 将保护第0页和第1页。 位1用于保护第2、3页(0x00080400至0x000807FF)。用户置1, 将允许对第2、3页进行写操作。清0, 将保护第2、3页。 位27用于保护第54、55页(0x86C00至0x86FFF)。用户置1, 将允许对第54、55页进行写操作。清0, 将保护第54、55页。 位28用于保护第56、57页(0x87000至0x873FF)。用户置1, 将允许对第56、57页进行写操作。清0, 将保护第56、57页。 |

临时保护

临时保护可通过直接写入FEEHID MMR来设置和取消。此寄存器为易失性, 因此保护只在器件上电时才能维持。此保护设置在周期供电后便不再重新载入。

加密永久性保护

加密永久性保护可通过FEEPRO设置以锁定保护配置。所需FEEPRO写入序列开头使用的软件密钥仅保存一次, 必须在以后的任何FEEHID或FEEPRO MMR访问中使用。批量擦除可将软件保护密钥设回到0xFFFF, 但同时也擦除了整个用户代码空间。

永久性保护

永久性保护可通过FEEPRO设置，设置方式类似于加密永久性保护，唯一不同在于所用软件密钥为0xDEADDEAD。FEEPRO写入序列保存后，只有批量擦除可将软件保护密钥设回到0xFFFFFFFF。这也会擦除整个用户代码空间。

写入软件保护密钥和设置永久性保护的序列

1. 写入和欲保护页相对应的FEEPRO。
2. 将新的(用户定义)32位软件保护密钥写入FEEADR [31:16]和FEEDAT的位[15:0]。
3. 将10写入FEEMOD的位[6:5]，并将FEEMOD的位3置1。
4. 执行FEECON内的保护命令(代码0x0C)。

要取消或修改保护，可使用相同的序列，同时修改FEEPRO值。

上述写入密钥并设置永久性保护的序列示例如下，此示例为Flash/EE的页4和页5提供写保护。

```
Int a = FEESTA;           // Ensure FEESTA
is cleared
FEEPRO = 0xFFFFFFFFB;    // Protect Page 4
and Page 5
FEEADR = 0x66BB;         // 32-bit key
value (Bits[31:16])
FEEDAT = 0xAA55;         // 32-bit key
value (Bits[15:0])
FEEMOD = 0x0048          // Lock security
sequence
FEECON = 0x0C;           // Write key
command
while (FEESTA & 0x04){} // Wait for
command to finish
```

批量擦除命令执行序列

```
FEEDAT = 0x3CFF;
FEEADR = 0xFFC3;
FEEMOD = FEEMOD|0x8;    //Erase key enable
FEECON = 0x06;          //Mass erase command
```

ADuC7060/ADuC7061

存储器映射寄存器

存储器映射寄存器(寄存器)空间被映射到存储器阵列的最上方两页。我们可以通过对ARM7寄存器组的间接寻址来存取存储器映射寄存器内的信息。

寄存器空间为CPU和所有片内外设提供接口。除了内核寄存器，所有的寄存器都位于寄存器区域内。图12内的阴影区域为未占用区域或保留区域，不允许用户程序访问该区域。表12为所有寄存器存储器映射。

读取或写入一个寄存器所需的存取时间取决于高级微控制器总线结构(AMBA)总线，该总线可用来访问外围设备。处理器有两个AMBA总线：高性能总线(AHB)用于系统模块，高级外围总线(APB)用于低性能外围设备。访问AHB需要一个周期，访问APB则需要两个周期。除了Flash/EE存储器、通用输入输出端口和PWM以外，ADuC7060/ADuC7061中的所有外围设备均位于APB上。

| | |
|------------|--------------------------------|
| 0xFFFFFFFF | |
| 0xFFFF0FC0 | |
| 0xFFFF0F80 | PWM |
| 0xFFFF0E24 | FLASH CONTROL INTERFACE |
| 0xFFFF0E00 | |
| 0xFFFF0D50 | |
| 0xFFFF0D00 | GPIO |
| 0xFFFF0A14 | |
| 0xFFFF0A00 | SPI |
| 0xFFFF0948 | |
| 0xFFFF0900 | I ² C |
| 0xFFFF0730 | |
| 0xFFFF0700 | UART |
| 0xFFFF0620 | |
| 0xFFFF0600 | DAC |
| 0xFFFF0570 | |
| 0xFFFF0500 | ADC |
| 0xFFFF0490 | |
| 0xFFFF048C | BAND GAP REFERENCE |
| 0xFFFF0470 | |
| 0xFFFF0450 | SPI/I ² C SELECTION |
| 0xFFFF0420 | |
| 0xFFFF0404 | PLL AND OSCILLATOR CONTROL |
| 0xFFFF0394 | |
| 0xFFFF0380 | GENERAL-PURPOSE TIMER |
| 0xFFFF0370 | |
| 0xFFFF0360 | WATCHDOG TIMER |
| 0xFFFF0350 | |
| 0xFFFF0340 | WAKE-UP TIMER |
| 0xFFFF0334 | |
| 0xFFFF0320 | GENERAL-PURPOSE TIMER |
| 0xFFFF0238 | |
| 0xFFFF0220 | REMAP AND SYSTEM CONTROL |
| 0xFFFF0140 | |
| 0xFFFF0000 | INTERRUPT CONTROLLER |

07079-007

图12存储器映射寄存器

完整的寄存器列表

在下列寄存器列表中，采用16进制码来表示地址。访问类型：R表示读，W表示写，R/W表示读写。

表17. IRQ基地址 = 0xFFFF0000

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------|-----|------|------------|---|
| 0x0000 | IRQSTA | 4 | R | 0x00000000 | 活动IRQ源状态。 |
| 0x0004 | IRQSIG | 4 | R | | 所有IRQ源的当前状态(使能和禁用)。 |
| 0x0008 | IRQEN | 4 | R/W | 0x00000000 | 使能的IRQ源。 |
| 0x000C | IRQCLR | 4 | W | 0x00000000 | 寄存器禁用IRQ源。 |
| 0x0010 | SWICFG | 4 | W | 0x00000000 | 软件中断配置寄存器。 |
| 0x0014 | IRQBASE | 4 | R/W | 0x00000000 | 所有矢量的基地址。指向64字节存储块的起始地址的指针；最多可以包含32个指向不同例行处理子程序的指针。 |
| 0x001C | IRQVEC | 4 | R | 0x00000000 | 该寄存器内包含与当前活动IRQ源对应的例行处理子程序的地址。 |
| 0x0020 | IRQP0 | 4 | R/W | 0x00000000 | 用于设置中断源1至中断源7的中断优先级。中断优先级可被设置为0至7。 |
| 0x0024 | IRQP1 | 4 | R/W | 0x00000000 | 设置中断源8至中断源15的中断优先级。 |
| 0x0028 | IRQP2 | 4 | R/W | 0x00000000 | 设置中断源16至中断源19的中断优先级。 |
| 0x0030 | IRQCONN | 4 | R/W | 0x00000000 | 使能IRQ和FIQ中断嵌套。 |
| 0x0034 | IRQCONE | 4 | R/W | 0x00000000 | 将外部中断源设置为上升沿/下降沿/电平触发。 |
| 0x0038 | IRQCLRE | 4 | R/W | 0x00000000 | 清除边沿/电平触发中断源。 |
| 0x003C | IRQSTAN | 4 | R/W | 0x00000000 | 最新导致中断异常的中断的优先级。 |
| 0x0100 | FIQSTA | 4 | R | 0x00000000 | 活动FIQ源状态。 |
| 0x0104 | FIQSIG | 4 | R | | 所有FIQ源的当前状态(使能和禁用)。 |
| 0x0108 | FIQEN | 4 | R/W | 0x00000000 | 使能的FIQ源。 |
| 0x010C | FIQCLR | 4 | W | 0x00000000 | 寄存器禁用FIQ源。 |
| 0x011C | FIQVEC | 4 | R | 0x00000000 | 该寄存器内包含与当前活动FIQ源对应的例行处理子程序的地址。 |
| 0x013C | FIQSTAN | 4 | R/W | 0x00000000 | 最新导致FIQ异常的FIQ的优先级。 |

表18. 系统控制基地址 = 0xFFFF0200

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|--------------------|-----|------|------|-----------------------|
| 0x0220 | REMAP ¹ | 1 | R/W | 0x00 | 重映射控制寄存器。参见“重映射操作”部分。 |
| 0x0230 | RSTSTA | 1 | R/W | 0x01 | RSTSTA状态寄存器。参见“复位”部分。 |
| 0x0234 | RSTCLR | 1 | W | 0x00 | 该寄存器用于清除RSTSTA寄存器。 |

¹ 由内核更新。

ADuC7060/ADuC7061

表19. 定时器基地址 = 0xFFFF0300

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|--------|-----|------|------------|--------------|
| 0x0320 | T0LD | 4 | R/W | 0x00000000 | 定时器0载入寄存器。 |
| 0x0324 | T0VAL | 4 | R | 0xFFFFFFFF | 定时器0值寄存器。 |
| 0x0328 | T0CON | 4 | R/W | 0x01000000 | 定时器0控制寄存器。 |
| 0x032C | T0CLRI | 1 | W | N/A | 定时器0中断清除寄存器。 |
| 0x0330 | T0CAP | 4 | R | 0x00000000 | 定时器0捕捉寄存器。 |
| 0x0340 | T1LD | 4 | R/W | 0x00000000 | 定时器1载入寄存器。 |
| 0x0344 | T1VAL | 4 | R | 0xFFFFFFFF | 定时器1值寄存器。 |
| 0x0348 | T1CON | 2 | R/W | 0x0000 | 定时器1控制寄存器。 |
| 0x034C | T1CLRI | 1 | W | N/A | 定时器1中断清除寄存器。 |
| 0x0360 | T2LD | 2 | R/W | 0x3BF8 | 定时器2载入寄存器。 |
| 0x0364 | T2VAL | 2 | R | 0x3BF8 | 定时器2值寄存器。 |
| 0x0368 | T2CON | 2 | R/W | 0x0000 | 定时器2控制寄存器。 |
| 0x036C | T2CLRI | 1 | W | N/A | 定时器2中断清除寄存器。 |
| 0x0380 | T3LD | 2 | R/W | 0x0000 | 定时器3载入寄存器。 |
| 0x0384 | T3VAL | 2 | R | 0xFFFF | 定时器3值寄存器。 |
| 0x0388 | T3CON | 4 | R/W | 0x00000000 | 定时器3控制寄存器。 |
| 0x038C | T3CLRI | 1 | W | N/A | 定时器3中断清除寄存器。 |
| 0x0390 | T3CAP | 2 | R | 0x0000 | 定时器3捕捉寄存器。 |

表20. PLL基地址 = 0xFFFF0400

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------|-----|------|--------|---|
| 0x0404 | POWKEY1 | 2 | W | 0xFFFF | POWCON0预写密钥。 |
| 0x0408 | POWCON0 | 1 | R/W | 0x7B | 电源控制和内核速度控制寄存器。 |
| 0x040C | POWKEY2 | 2 | W | 0xFFFF | POWCON0写后密钥。 |
| 0x0410 | PLLKEY1 | 2 | W | 0xFFFF | PLLCON预写密钥。 |
| 0x0414 | PLLCON | 1 | R/W | 0x00 | PLL时钟源选择寄存器。 |
| 0x0418 | PLLKEY2 | 2 | W | 0xFFFF | PLLCON写后密钥。 |
| 0x0434 | POWKEY3 | 2 | W | 0xFFFF | POWCON1预写密钥。 |
| 0x0438 | POWCON1 | 2 | R/W | 0x124 | 电源控制寄存器。 |
| 0x043C | POWKEY4 | 2 | W | 0xFFFF | POWCON1写后密钥。 |
| 0x0464 | GPOKEY1 | 2 | W | 0xFFFF | GPOCON1预写密钥。 |
| 0x0468 | GPOCON1 | 1 | R/W | 0x00 | 将P0.0、P0.1、P0.2和P0.3设置为模拟输入端或数字输入/输出端。此外，可使能SPI或I ² C模式。 |
| 0x046C | GPOKEY2 | 2 | W | 0xFFFF | GPOCON1写后密钥。 |

表21. ADC基地址 = 0xFFFF0500

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------------------|-----|------|-----------------------|--|
| 0x0500 | ADCSTA | 2 | R | 0x0000 | ADC状态寄存器。 |
| 0x0504 | ADCMSKI | 2 | R/W | 0x0000 | ADC中断源使能寄存器。 |
| 0x0508 | ADCMDDE | 1 | R/W | 0x03 | ADC模式寄存器。 |
| 0x050C | ADC0CON | 2 | R/W | 0x8000 | 主ADC控制寄存器。 |
| 0x0510 | ADC1CON | 2 | R/W | 0x0000 | 辅助ADC控制寄存器。 |
| 0x0514 | ADCFLT | 2 | R/W | 0x0007 | ADC滤波器控制寄存器。 |
| 0x0518 | ADCCFG | 1 | R/W | 0x00 | ADC配置寄存器。 |
| 0x051C | ADC0DAT | 4 | R | 0x00000000 | 主ADC结果寄存器。 |
| 0x0520 | ADC1DAT | 4 | R | 0x00000000 | 辅助ADC结果寄存器。 |
| 0x0524 | ADC0OF ¹ | 2 | R/W | 0x0000, 特定器件, 工厂编程 | 主ADC失调校准设置。 |
| 0x0528 | ADC1OF ¹ | 2 | R/W | 0x0000; 特定器件, 工厂编程 | 辅助ADC失调寄存器。 |
| 0x052C | ADC0GN ¹ | 2 | R/W | 0x5555 | 主ADC失调寄存器。 |
| 0x0530 | ADC1GN ¹ | 2 | R/W | 0x5555 | 辅助ADC失调寄存器。见表42中对ADC操作模式配置位VA (ADCLPMCFCFG[1:0])的说明。 |
| 0x0534 | ADC0RCR | 2 | R/W | 0x0001 | 主ADC结果计数器/重载寄存器。 |
| 0x0538 | ADC0RCV | 2 | R | 0x0000 | 主ADC结果计数器寄存器。 |
| 0x053C | ADC0TH | 2 | R/W | 0x0000 | 主ADC 16位比较器阈值寄存器。 |
| 0x0540 | ADC0THC | 2 | R/W | 0x0001 | 主ADC 16位比较器阈值计数器限值。 |
| 0x0544 | ADC0THV | 2 | R | 0x0000 | ADC0 8位阈值超过计数器寄存器 |
| 0x0548 | ADC0ACC | 4 | R | 0x00000000 | 主ADC累加器。 |
| 0x054C | ADC0ATH | 4 | R/W | 0x00000000 | 主ADC 32位比较器阈值寄存器。 |
| 0x0570 | IEXCON | 1 | R/W | 0x00 | 激励电流源控制寄存器。 |

¹ 被内核更新为特定器件校准值。

表22. DAC控制基地址 = 0xFFFF0600

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------|-----|------|------------|-------------|
| 0x0600 | DAC0CON | 2 | R/W | 0x0200 | DAC控制寄存器。 |
| 0x0604 | DAC0DAT | 4 | R/W | 0x00000000 | DAC输出数据寄存器。 |

表23. UART基地址 = 0xFFFF0700

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------|-----|------|--------|------------------|
| 0x0700 | COMTX | 1 | W | N/A | UART发送寄存器。 |
| 0x0700 | COMRX | 1 | R | 0x00 | UART接收寄存器。 |
| 0x0700 | COMDIV0 | 1 | R/W | 0x00 | UART标准波特率发生器除数0。 |
| 0x0704 | COMIEN0 | 1 | R/W | 0x00 | UART中断使能寄存器0。 |
| 0x0704 | COMDIV1 | 1 | R/W | 0x00 | UART标准波特率发生器除数1。 |
| 0x0708 | COMIID0 | 1 | R | 0x01 | UART中断标识0。 |
| 0x070C | COMCON0 | 1 | R/W | 0x00 | UART控制寄存器0。 |
| 0x0710 | COMCON1 | 1 | R/W | 0x00 | UART控制寄存器1。 |
| 0x0714 | COMSTA0 | 1 | R | 0x60 | UART状态寄存器0。 |
| 0x0718 | COMSTA1 | 1 | R | 0x00 | UART状态寄存器1。 |
| 0x072C | COMDIV2 | 2 | R/W | 0x0000 | UART小数分频器寄存器。 |

ADuC7060/ADuC7061

表24. I²C基地址 = 0xFFFF0900

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|----------|-----|------|--------|--|
| 0x0900 | I2CMCON | 2 | R/W | 0x0000 | I ² C主机控制寄存器。 |
| 0x0904 | I2CMSTA | 2 | R | 0x0000 | I ² C主机状态寄存器。 |
| 0x0908 | I2CMRX | 1 | R | 0x00 | I ² C主机接收寄存器。 |
| 0x090C | I2CMTX | 1 | W | 0x00 | I ² C主机发送寄存器。 |
| 0x0910 | I2CMCNT0 | 2 | R/W | 0x0000 | I ² C主机读计数寄存器。在从从机设备中读取数据之前，须向该寄存器写入需要读取的字节数。 |
| 0x0914 | I2CMCNT1 | 1 | R | 0x00 | I ² C主机当前读取计数寄存器。该寄存器用于记录已从从机设备中接收的字节数。 |
| 0x0918 | I2CADR0 | 1 | R/W | 0x00 | 地址字节寄存器。在与从机进行通信前，向该寄存器内写入目标从机地址。 |
| 0x091C | I2CADR1 | 1 | R/W | 0x00 | 地址字节寄存器。在与从机进行通信前，向该寄存器内写入目标从机地址。只可用于10位模式。 |
| 0x0924 | I2CDIV | 2 | R/W | 0x1F1F | I ² C时钟控制寄存器。用于设置SCLK频率。 |
| 0x0928 | I2CSCON | 2 | R/W | 0x0000 | I ² C从机控制寄存器。 |
| 0x092C | I2CSSTA | 2 | R/W | 0x0000 | I ² C从机状态寄存器。 |
| 0x0930 | I2CSRX | 1 | R | 0x00 | I ² C从机接收寄存器。 |
| 0x0934 | I2CSTX | 1 | W | 0x00 | I ² C从机发送寄存器。 |
| 0x0938 | I2CALT | 1 | R/W | 0x00 | I ² C硬件广播识别寄存器。 |
| 0x093C | I2CID0 | 1 | R/W | 0x00 | I ² C从机识别号0寄存器。从机总线识别号寄存器。 |
| 0x0940 | I2CID1 | 1 | R/W | 0x00 | I ² C从机识别号1寄存器。从机总线识别号寄存器。 |
| 0x0944 | I2CID2 | 1 | R/W | 0x00 | I ² C从机识别号2寄存器。从机总线识别号寄存器。 |
| 0x0948 | I2CID3 | 1 | R/W | 0x00 | I ² C从机识别号3寄存器。从机总线识别号寄存器。 |
| 0x094C | I2CFSTA | 2 | R/W | 0x0000 | I ² C FIFO状态寄存器。可应用于主机模式和从机模式。 |

表25. SPI基地址 = 0xFFFF0A00

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|--------|-----|------|------------|--------------|
| 0x0A00 | SPISTA | 4 | R | 0x00000000 | SPI状态寄存器。 |
| 0x0A04 | SPIRX | 1 | R | 0x00 | SPI接收寄存器。 |
| 0x0A08 | SPLITX | 1 | W | 0x00 | SPI发送寄存器。 |
| 0x0A0C | SPIDIV | 1 | W | 0x1B | SPI波特率选择寄存器。 |
| 0x0A10 | SPICON | 2 | R/W | 0x0000 | SPI控制寄存器。 |

表26. GPIO基地址 = 0xFFFF0D00

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|---------|-----|------|------------|-------------------|
| 0x0D00 | GP0CON0 | 4 | R/W | 0x00000000 | 通用输入输出端口0控制寄存器。 |
| 0x0D04 | GP1CON | 4 | R/W | 0x00000000 | 通用输入输出端口1控制寄存器。 |
| 0x0D08 | GP2CON | 4 | R/W | 0x00000000 | 通用输入输出端口2控制寄存器。 |
| 0x0D20 | GP0DAT | 4 | R/W | 0x000000XX | 通用输入输出端口0数据控制寄存器。 |
| 0x0D24 | GP0SET | 4 | W | 0x000000XX | 通用输入输出端口0数据设置寄存器。 |
| 0x0D28 | GP0CLR | 4 | W | 0x000000XX | 通用输入输出端口0数据清除寄存器。 |
| 0x0D2C | GP0PAR | 4 | R/W | 0x00000000 | 通用输入输出端口0上拉禁用寄存器。 |
| 0x0D30 | GP1DAT | 4 | R/W | 0x000000XX | 通用输入输出端口1数据控制寄存器。 |
| 0x0D34 | GP1SET | 4 | W | 0x000000XX | 通用输入输出端口1数据设置寄存器。 |
| 0x0D38 | GP1CLR | 4 | W | 0x000000XX | 通用输入输出端口1数据清除寄存器。 |
| 0x0D3C | GP1PAR | 4 | R/W | 0x00000000 | 通用输入输出端口1上拉禁用寄存器。 |
| 0x0D40 | GP2DAT | 4 | R/W | 0x000000XX | 通用输入输出端口2数据控制寄存器。 |
| 0x0D44 | GP2SET | 4 | W | 0x000000XX | 通用输入输出端口2数据设置寄存器。 |
| 0x0D48 | GP2CLR | 4 | W | 0x000000XX | 通用输入输出端口2数据清除寄存器。 |
| 0x0D4C | GP2PAR | 4 | R/W | 0x00000000 | 通用输入输出端口2上拉禁用寄存器。 |

表27. Flash/EE基地址 = 0xFFFF0E00

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|--------|-----|------|------------|--------------------|
| 0x0E00 | FEESTA | 2 | R | 0x20 | Flash/EE状态寄存器。 |
| 0x0E04 | FEEMOD | 2 | R/W | 0x0000 | Flash/EE控制寄存器。 |
| 0x0E08 | FEECON | 1 | R/W | 0x07 | Flash/EE控制寄存器。 |
| 0x0E0C | FEEDAT | 2 | R/W | 0xFFFF | Flash/EE数据寄存器。 |
| 0x0E10 | FEEADR | 2 | R/W | 0x0000 | Flash/EE地址寄存器。 |
| 0x0E18 | FEESIG | 3 | R | 0xFFFFFFFF | Flash/EE LFSR 寄存器。 |
| 0x0E1C | FEEPRO | 4 | R/W | 0x00000000 | Flash/EE保护寄存器。 |
| 0x0E20 | FEEHID | 4 | R/W | 0xFFFFFFFF | Flash/EE保护寄存器。 |

表28. PWM基地址 = 0xFFFF0F80

| 地址 | 名称 | 字节数 | 访问类型 | 默认值 | 描述 |
|--------|----------|-----|------|--------|---------------------------------|
| 0x0F80 | PWMCON | 2 | R/W | 0x0012 | PWM控制寄存器。详细的说明请参见“脉宽调制器”部分。 |
| 0x0F84 | PWM0COM0 | 2 | R/W | 0x0000 | 用于比较PWM输出0和PWM输出1的比较寄存器0。 |
| 0x0F88 | PWM0COM1 | 2 | R/W | 0x0000 | 用于比较PWM输出0和PWM输出1的比较寄存器1。 |
| 0x0F8C | PWM0COM2 | 2 | R/W | 0x0000 | 用于比较PWM输出0和PWM输出1的比较寄存器2。 |
| 0x0F90 | PWM0LEN | 2 | R/W | 0x0000 | 用于PWM输出0和PWM输出1的频率控制 |
| 0x0F94 | PWM1COM0 | 2 | R/W | 0x0000 | 用于比较PWM输出2和PWM输出3的比较寄存器0。 |
| 0x0F98 | PWM1COM1 | 2 | R/W | 0x0000 | 用于比较PWM输出2和PWM输出3的比较寄存器1。 |
| 0x0F9C | PWM1COM2 | 2 | R/W | 0x0000 | 用于比较PWM输出2和PWM输出3的比较寄存器2。 |
| 0x0FA0 | PWM1LEN | 2 | R/W | 0x0000 | 用于PWM输出2和PWM输出3的频率控制 |
| 0x0FA4 | PWM2COM0 | 2 | R/W | 0x0000 | 用于比较PWM输出4和PWM输出5的比较寄存器0。 |
| 0x0FA8 | PWM2COM1 | 2 | R/W | 0x0000 | 用于比较PWM输出4和PWM输出5的比较寄存器1。 |
| 0x0FAC | PWM2COM2 | 2 | R/W | 0x0000 | 用于比较PWM输出4和PWM输出5的比较寄存器2。 |
| 0x0FB0 | PWM2LEN | 2 | R/W | 0x0000 | 用于PWM输出4和PWM输出5的频率控制。 |
| 0x0FB8 | PWMCLRI | 2 | W | 0x0000 | PWM中断清除寄存器。向其中写入任意值就可以清除PWM中断源。 |

ADuC7060/ADuC7061

复位

有四种复位形式：外部复位、上电复位、看门狗复位及软件复位。RSTSTA寄存器指示产生上一次复位的复位源；可通过用户代码对该寄存器进行写操作，以启动软件复位事件。

此寄存器的清零，可通过对位于0xFFFF0234地址处的RSTCLR寄存器进行写操作来完成。RSTCLR的位分配镜像RSTSTA的位分配。这两种寄存器用于在复位异常服务例行程序运行期间识别复位源。四种复位事件的含义如表30所示。

RSTSTA寄存器

名称： RSTSTA
 地址： 0xFFFF0230
 默认值： 因复位类型而异
 访问类型： 读、写
 功能： 此8位寄存器指示上一次复位事件的复位源；可由用户代码对该寄存器进行写操作，以启动软件复位。

RSTCLR寄存器

名称： RSTCLR
 地址： 0xFFFF0234
 访问类型： 只写
 功能： 此8位只写寄存器可将RSTSTA的对应位清零。

表29. RSTSTA/RSTCLR 寄存器位功能描述

| 位 | 描述 |
|-----|---|
| 7:4 | 未使用。全部不用，始终读取0。 |
| 3 | 外部复位。 发生外部复位时，该位自动置1。 通过设置RSTCLR寄存器内的相应位，可将该位清0。 |
| 2 | 软件复位。 可通过用户代码将该位置1，以产生软件复位。 通过设置RSTCLR寄存器内的相应位，可将该位清0。 ¹ |
| 1 | 看门狗超时。 发生看门狗超时时，该位自动置1。 通过设置RSTCLR寄存器内的相应位，可将该位清0。 |
| 0 | 上电复位。 发生上电复位时，该位自动置位。 通过设置RSTCLR寄存器内的相应位，可将该位清0。 |

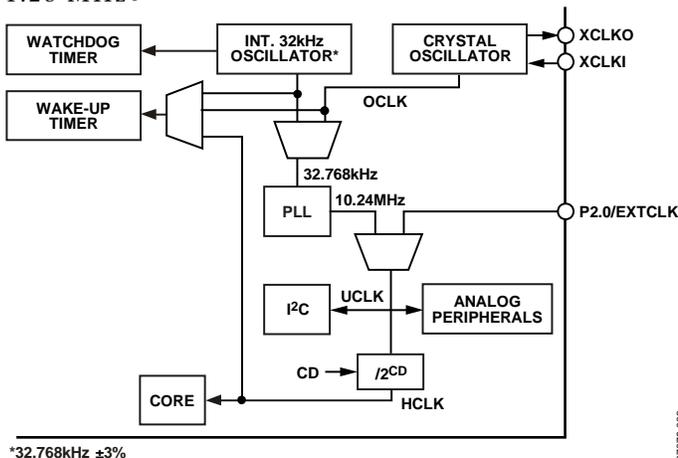
¹ 如果RSTSTA的软件复位位已设，任何不将此位清0的RSTCLR写入操作都会产生软件复位。

表30. 器件复位含义

| 复位 | 将外部引脚复位到默认状态 | 内核执行 | 复位所有外部寄存器 (不包括RSTSTA) | 复位外设 | 复位看门狗定时器 | RAM有效 | RSTSTA状态 (复位事件后) |
|------|--------------|------|-----------------------|------|----------|-------|------------------|
| POR | 是 | 是 | 是 | 是 | 是 | 是/否 | RSTSTA[0] = 1 |
| 看门狗 | 是 | 是 | 是 | 是 | 否 | 是 | RSTSTA[1] = 1 |
| 软件 | 是 | 是 | 是 | 是 | 否 | 是 | RSTSTA[2] = 1 |
| 外部引脚 | 是 | 是 | 是 | 是 | 否 | 是 | RSTSTA[3] = 1 |

振荡器、锁相环和电源控制 时钟系统

ADuC7060/ADuC7061系列产品内部集成了一个32.768 kHz振荡器(误差在±3%范围内)、一个时钟分频器和一个锁相环(PLL)。PLL将内部振荡器或一个外部32.768 KHz晶振的倍数频率锁定,为系统提供一个稳定的10.24 MHz时钟(UCLK)。为了省电,内核可以工作在这个频率下,或这个频率的 $1/2^n$ 。内核的实际工作频率 $UCLK/2^{CD}$,称为HCLK。默认的内核时钟分频为8分频($CD=3$),或1.28 MHz。



*32.768kHz ±3%

图13.时钟系统

外部晶振选择

在用外部晶振时,用户必须遵循如下程序:

1. 使能Timer1中断,并将其超时周期设置为大于120 μs。
2. 根据PLLCON寄存器的写时序要求,将OSEL位设置为[10]且清除EXTCLK位。
3. 按照POWCON寄存器的正确写时序,强制器件进入浅休眠模式。
4. 当器件从浅休眠模式被定时器1中断源唤醒后,时钟源已经切换到外部晶振了。

源代码示例

```
T1LD = 0x80; // 32,768 clock ticks
T1CON = 0xC0; // Periodic mode, enable
           // timer, 32,768 Hz clock/1
IRQEN |= 0x10; // Enable Timer1 interrupt
           // source
PLLKEY1 = 0xAA; // Switch to external crystal
PLLCON = 0x2;
PLLKEY2 = 0x55;

POWKEY1 = 0x1; // Enter nap mode
POWCON0 = 0x73;
POWKEY2 = 0xF4;
```

为了避免晶振失振,应该使用看门狗定时器。在初始化期间,通常应对RSTSTA进行测试,以判断是否有来自看门狗定时器的复位信号。

外部时钟选择

如果需要使用外部时钟,必须将P2.0配置为模式0。外部时钟频率最高可达20.48 MHz,容差为1%。外部时钟在器件内部被2分频。

源代码示例

```
T1LD = 0x80;
T1CON = 0xC0;
IRQEN |= 0x10; // Enable Timer1 interrupt
PLLKEY1 = 0xAA; // Switch to external clock
PLLCON = 0x4;
PLLKEY2 = 0x55;

POWKEY1 = 0x1; // Enter NAP mode
POWCON0 = 0x73;
POWKEY2 = 0xF4;
```

时钟源的选择是由PLLCON寄存器控制的。默认情况下,选用内部振荡器作为PLL的输入。

电源控制系统

通过写POWCON0寄存器,可以改变内核时钟频率。这是一个重要的保护寄存器;因此,在设置POWCON0寄存器之前和之后都必须立即对寄存器POWKEY1和POWKEY2进行写操作。下面将举例说明如何将内核时钟频率设置为10.24 MHz:

```
POWKEY1 = 0x1;
POWCON0 = 0x78; //Set core to max CPU
                //speed of 10.24 MHz
POWKEY2 = 0xF4;
```

ADuC7060/ADuC7061提供了多种工作模式。表33描述了各个模式下上电的器件及上电时间。

表34给出了在不同时钟分频位决定的各模式的总的电流(模拟电源电流+数字电源电流)消耗值。ADC处于关闭状态。注意:这些测量值包括调节器和测试板上其他器件消耗的电流。

ADuC7060/ADuC7061

通过对POWCON1进行写操作，可以关断UART、PWM或I²C/SPI模块，从而降低器件在活动状态下的功耗。如需访问POWCON1，必须在访问前首先通过指令将POWKEY3设为0x76，在访问POWCON1后，应立即通过指令将POWKEY4设为0xB1。

例如，运行下列代码，可使能SPI/I²C模块，但也会关断PWM和UART模块。

```
POWKEY3 =0x76;
POWCON1 =0x4;           //0x100 PWM; 0x20
Uart; 0x4 SPI/I2C
POWKEY4 =0xB1;
```

电源和时钟控制寄存器

POWCON1寄存器

名称: POWKEY1
地址: 0xFFFF0404
默认值: 0xFFFF
访问类型: 写
功能: 如需向POWCON0写入数据，必须首先向该寄存器内写入0x01。

POWCON0寄存器

名称: POWCON0
地址: 0xFFFF0408
默认值: 0x7B
访问类型: 读、写
功能: 该寄存器可控制用于控制CPU时钟(HCLK)的时钟分频位。

表31. POWCON0 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|---------|---|
| 7 | 保留 | 该位的值必须始终为0 |
| 6 | XPD | XTAL关断。 用户清0，关断外部晶体电路。 用户置1，使能外部晶体电路。 |
| 5 | PLLPD | PLL关断。如果定时器外设由PLL输出时钟驱动，则关断。如果由有效时钟源驱动，则保持正常功耗模式。 该位清0，关断PLL。当启用内核或外设后，不允许关断PLL；位3、位4、位5必须同时被清零。 默认置1，且在产生唤醒事件时由硬件置1。 |
| 4 | PPD | 外设关断。该位可以关断的外设有： SRAM、Flash/EE存储器、GPIO接口、SPI/I ² C和UART串行端口。 清0，关断外设。启用内核后，不允许关断外设；位3、位4必须同时被清零。 默认置1和/或当产生唤醒事件时，由硬件置1。唤醒定时器(Timer1)可继续保持活动状态。 |
| 3 | COREPD | 内核关断。通过用户代码将MCU关断时，在将关断命令写入POWCON0后插入一个假MCU周期。 清0，关断ARM内核。 默认置1，且当产生唤醒事件时，由硬件置1。 |
| 2:0 | CD[2:0] | 内核时钟取决于CD的设置。 [000] = 10.24 MHz [001] = 5.12 MHz [010] = 2.56 MHz [011] = 1.28 MHz [默认值] [100] = 640 kHz [101] = 320 kHz [110] = 160 kHz [111] = 80 kHz |

POWKEY2寄存器

名称: POWKEY2
 地址: 0xFFFF040C
 默认值: 0xFFFF
 访问类型: 写
 功能: 如需向POWCON0寄存器写入数据, 必须首先通过指令向该寄存器内写入0xF4。

POWCON1寄存器

名称: POWKEY1
 地址: 0xFFFF0438
 默认值: 0x124
 访问类型: 读/写
 功能: 此寄存器控制PWM、UART和I2C/SPI模块的时钟信号。
 禁用这些模块的时钟可降低功耗。

POWKEY3寄存器

名称: POWKEY3
 地址: 0xFFFF0434
 默认值: 0xFFFF
 访问类型: 写
 功能: 如需向POWCON1寄存器写入数据, 必须首先通过指令向该寄存器内写入0x76。

POWKEY4寄存器

名称: POWKEY4
 地址: 0xFFFF043C
 默认值: 0xFFFF
 访问类型: 写
 功能: 如需向POWCON1寄存器写入数据, 必须首先通过指令向该寄存器内写入0xB1。

表32. POWCON1 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|-----------|--|
| 15:9 | 保留 | 该位的值必须始终为0。 |
| 8 | PWMOFF | PWM关断位。 用户置1, 可启用PWM模块。该位默认置1。 用户清0, 可关断PWM模块。 |
| 7:6 | 保留 | 保留位。始终将这些位清0。 |
| 5 | UARTOFF | UART关断位。 用户置1, 可启用UART模块。该位默认置1。 用户清0, 可关断UART模块。 |
| 4:3 | 保留 | 保留位。始终将这些位清0。 |
| 2 | I2CSPIOFF | I2C/SPI关断位。 用户置1, 可启用I2C/SPI模块。该位默认置1。 用户清0, 可关断I2C/SPI模块。 |
| 1:0 | 保留 | 保留位。始终将这些位清0。 |

表33. ADuC7060/ADuC7061省电模式

| POWCON0[6:3] | 模式 | 内核 | 外设 | PLL | XTAL/T1/T2 | IRQ0至IRQ3 | 启动/上电时间 |
|--------------|------------|----|----|-----|------------|-----------|--------------------------------|
| 1111 | 活动(Active) | 是 | 是 | 是 | 是 | 是 | 130 ms(CD = 0) |
| 1110 | 暂停(Pause) | | 是 | 是 | 是 | 是 | 4.8 μs(CD = 0); 660 μs(CD = 7) |
| 1100 | 浅休眠(NAP) | | | 是 | 是 | 是 | 4.8 μs(CD = 0); 660 μs(CD = 7) |
| 1000 | 睡眠(Sleep) | | | | 是 | 是 | 66 μs(CD = 0); 900 μs(CD = 7) |
| 0000 | 停止(Stop) | | | | | 是 | 66 μs(CD = 0); 900 μs(CD = 7) |

ADuC7060/ADuC7061

表34. 25°C时的典型功耗值¹(单位: mA)

| POWCON0[6:3] | 模式 | CD = 0 | CD = 1 | CD = 2 | CD = 3 | CD = 4 | CD = 5 | CD = 6 | CD = 7 |
|--------------|-------------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1111 | 活动(Active) ² | 5.22 | 4.04 | 2.69 | 2.01 | 1.67 | 1.51 | 1.42 | 1.38 |
| 1110 | 暂停(Pause) ³ | 2.6 | 1.95 | 1.6 | 1.49 | 1.4 | 1.33 | 1.31 | 1.3 |
| 1100 | 浅休眠(NAP) ³ | 1.33 | 1.29 | 1.29 | 1.29 | 1.29 | 1.29 | 1.29 | 1.29 |
| 1000 | 睡眠(Sleep) ³ | 0.085 | 0.085 | 0.085 | 0.085 | 0.085 | 0.085 | 0.085 | 0.085 |
| 0000 | 停止(Stop) ³ | 0.055 | 0.055 | 0.055 | 0.055 | 0.055 | 0.055 | 0.055 | 0.055 |

¹ 表34中列出的数值均为当两个ADC处于关闭状态下的数值。

² 在活动模式下, GPOPAR位7=1。

³ 表中列出的对应于暂停、浅休眠、睡眠和停止模式下的各个值均是在nTRST引脚为低电平时测量所得。为将在各种模式下由nTRST引起的IDD最小化, 需将GPOPAR寄存器位7置1。这样, 可以禁用nTRST引脚的内部下拉电阻, 而且意味着nTRST引脚与外部上拉电阻之间没有接地通路。GPOPAR寄存器位7的默认值为0; 因此, 利用用户代码将该位置为1不会影响BM操作。

名称: PLLKEY1
 地址: 0xFFFF0410
 默认值: 0xXXXXX
 访问类型: 写
 功能: 如需向PLLCON写入数据, 必须首先向该寄存器内写入0xAA。

名称: PLLCON
 地址: 0xFFFF0414
 默认值: 0x00
 访问类型: 读、写
 功能: 该寄存器为PLL选择时钟输入。

表35. PLLCON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|--------|---|
| 7:3 | 保留 | 这些位总是清0。 |
| 2 | EXTCLK | 通过将该位置为1, 可从P2.0选择外部时钟输入。通过将该位清0, 可禁用外部时钟。 |
| 1:0 | OSEL | 振荡器选择位。 [00] = 内部32,768 Hz振荡器 [01] = 内部32,768 Hz振荡器 [10] = 外部晶振 [11] = 内部32,768 Hz振荡器 |

名称: PLLKEY2
 地址: 0xFFFF0418
 默认值: 0xXXXXX
 访问类型: 写
 功能: 在PLLCON写入数据后, 必须立即向该寄存器内写入0x55。

ADC电路信息

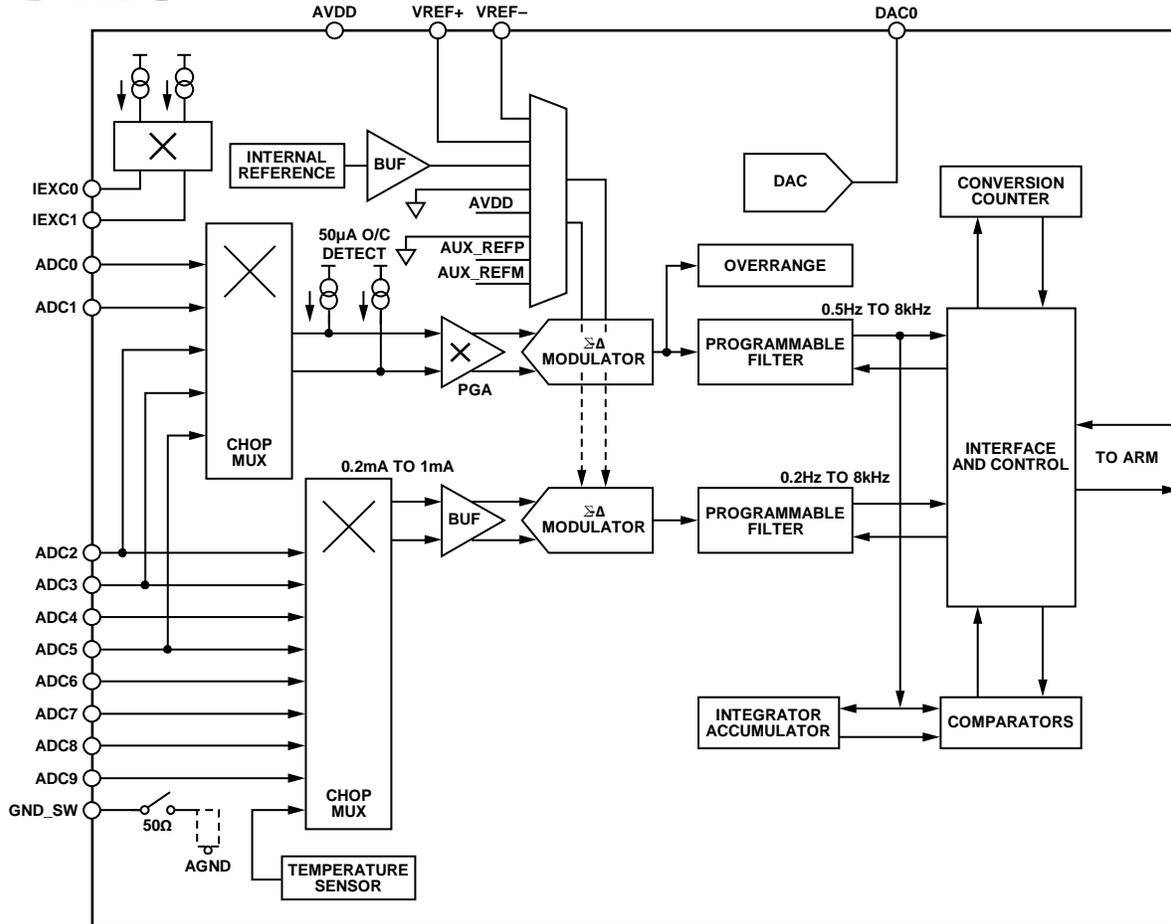


图14. 模拟模块框图

ADuC7060/ADuC7061集成有两个独立的多通道 Σ - Δ ADC。主ADC是一个24位4通道ADC。辅助ADC是一个24位 Σ - Δ ADC，最高可支持7个单端输入通道。

主ADC输入级有一个多路复用器和一个可编程增益放大器。位于主通道上的多路复用器可配置成两个全差分输入通道或者四个单端输入通道。

辅助ADC输入级内置了一个缓冲器。这两个ADC都具备数字滤波功能，可以在相当宽的动态范围内测量低频信号，

例如那些在压力传感器、温度传感器、电子秤或者应变计应用中产生的信号。

ADuC7060/ADuC7061系列产品的辅助ADC可配置成四个全差分输入通道或者七个单端输入通道。

由于具有内部缓冲器，内部通道可以直接转换从传感器传来的信号，而无需外接信号调理模块。

ADuC7060/ADuC7061

表36. 主ADC-在正常模式下的典型输出均方根噪声(单位: μV)¹

| ADC寄存器状态 | 数据更新速率 | 可选输入范围 (PGA设置) | | | | | | | | | |
|----------|--------|------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---------------------------------|----------------------------------|-----------------------------------|----------------------------------|----------------------------------|
| | | $\pm 1.2\text{ V}$ (PGA = 1) | $\pm 600\text{ mV}$ (PGA = 2) | $\pm 300\text{ mV}$ (PGA = 4) | $\pm 150\text{ mV}$ (PGA = 8) | $\pm 75\text{ mV}$ (PGA = 16) | $\pm 37.5\text{ mV}$ (PGA = 32) | $\pm 18.75\text{ mV}$ (PGA = 64) | $\pm 9.375\text{ mV}$ (PGA = 128) | $\pm 4.68\text{ mV}$ (PGA = 256) | $\pm 2.34\text{ mV}$ (PGA = 512) |
| 斩波开启 | 4 Hz | 0.62 μV | 0.648 μV | 0.175 μV | 0.109 μV | 0.077 μV | 0.041 μV | 0.032 μV | 0.0338 μV | 0.032 μV | 0.033 μV |
| 斩波关闭 | 50 Hz | 1.97 μV | 1.89 μV | 0.570 μV | 0.38 μV | 0.27 μV | 0.147 μV | 0.123 μV | 0.12 μV | 0.098 μV | 0.098 μV |
| 斩波关闭 | 1 kHz | 8.54 μV | 8.4 μV | 2.55 μV | 1.6 μV | 1.17 μV | 0.658 μV | 0.53 μV | 0.55 μV | 0.56 μV | 0.52 μV |
| 斩波关闭 | 8 kHz | 54.97 μV | 55.54 μV | 14.30 μV | 7.88 μV | 4.59 μV | 2.5 μV | 1.71 μV | 1.75 μV | 0.915 μV | 0.909 μV |

¹ 输入电压范围以共模电压为中心, 应符合“电气规格”部分规定的输入电压范围。

表37. 主ADC-在正常状态下典型输出均方根有效位数(括号内为峰峰位)

| ADC寄存器状态 | 数据更新速率 | 输入电压噪声(单位: mV) | | | | | | | | | |
|----------|--------|------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---------------------------------|----------------------------------|-----------------------------------|----------------------------------|----------------------------------|
| | | $\pm 1.2\text{ V}$ (PGA = 1) | $\pm 600\text{ mV}$ (PGA = 2) | $\pm 300\text{ mV}$ (PGA = 4) | $\pm 150\text{ mV}$ (PGA = 8) | $\pm 75\text{ mV}$ (PGA = 16) | $\pm 37.5\text{ mV}$ (PGA = 32) | $\pm 18.75\text{ mV}$ (PGA = 64) | $\pm 9.375\text{ mV}$ (PGA = 128) | $\pm 4.68\text{ mV}$ (PGA = 256) | $\pm 2.34\text{ mV}$ (PGA = 512) |
| 斩波开启 | 4 Hz | 21.9 (19.1 p-p) | 20.8 (18.1 p-p) | 21.7 (19.0 p-p) | 21.4 (18.7 p-p) | 20.9 (18.2 p-p) | 20.8 (18.1 p-p) | 20.2 (17.4 p-p) | 19.1 (16.4 p-p) | 18.2 (15.4 p-p) | 17.1 (14.4 p-p) |
| 斩波关闭 | 50 Hz | 20.2 (17.5 p-p) | 19.3 (16.6 p-p) | 20.0 (17.3 p-p) | 19.6 (16.9 p-p) | 19.1 (16.4 p-p) | 19.0 (16.2 p-p) | 18.2 (15.5 p-p) | 17.3 (14.6 p-p) | 16.6 (13.8 p-p) | 15.5 (12.8 p-p) |
| 斩波关闭 | 1 kHz | 18.1 (15.3 p-p) | 17.1 (14.4 p-p) | 17.8 (15.1 p-p) | 17.5 (14.8 p-p) | 17.0 (14.2 p-p) | 16.8 (14.1 p-p) | 16.1 (13.4 p-p) | 15.1 (12.3 p-p) | 14.0 (11.3 p-p) | 13.1 (10.4 p-p) |
| 斩波关闭 | 8 kHz | 15.4 (12.7 p-p) | 14.4 (11.7 p-p) | 15.4 (12.6 p-p) | 15.2 (12.5 p-p) | 15.0 (12.3 p-p) | 14.9 (12.2 p-p) | 14.4 (11.7 p-p) | 13.4 (10.7 p-p) | 13.3 (10.6 p-p) | 12.3 (9.6 p-p) |

表38. 辅助ADC-典型输出均方根噪声

| ADC寄存器 | 数据更新速率 | 均方根值 |
|--------|--------|---------------------|
| 斩波开启 | 4 Hz | 0.633 μV |
| 斩波开启 | 10 Hz | 0.810 μV |
| 斩波关闭 | 1 kHz | 7.4 μV |
| 斩波关闭 | 8 kHz | 54.18 μV |

基准电压源

主ADC和辅助ADC均可选择使用内部基准电压或者两个外部差分基准电压源之一。第一个外部基准电压源施加于VREF+/VREF-引脚。第二个外部基准电压源施加于ADC4/EXT_REF2IN+和ADC5/EXT_REF2IN-引脚。在默认状态下, ADC使用内部1.2 V基准电压源。

如需了解为主ADC设置外部基准电压源的详细说明, 请参阅对ADC0CON(ADC0控制寄存器的ADC0REF[1:0]位)的说明。

如需了解为辅助ADC设置外部基准电压源的详细说明, 请参阅对ADC1CON(ADC1控制寄存器的ADC1REF[2:0]位)的说明。

通过将ADC0CON寄存器的HIGHEXTREF0置位, 可以为ADC0提供一个大于1.35 V的外部基准电压源。

同样, 通过将ADC1CON寄存器的高EXTREF1置位, 也可以为ADC1提供一个大于1.35 V的外部基准电压源。

诊断电流源

为能够检测到与外部传感器的连接是否成功, ADuC7060/ADuC7061在主ADC和辅助ADC选中的模拟输入通道内集成了一个50 μA 的恒流源。

ADC0CON寄存器的ADC0DIAG[1:0]位用于控制主ADC模拟输入端的诊断电流源。

同样, ADC1CON寄存器的ADC1DIAG[1:0]位用于控制辅助ADC模拟输入的诊断电流源。

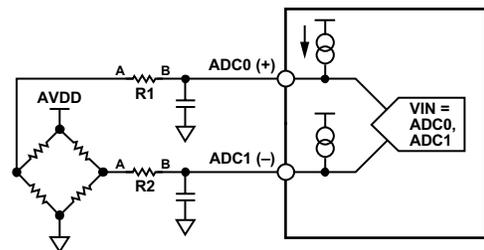


图15. 采用诊断电流源的电路示例

表39. 采用诊断电流源的示例情形

| 诊断测试 | | 正常结果 | 错误结果 | 检测到的测量故障 |
|-------------------|--|--|-------------------------------------|--------------------------------|
| 寄存器设置 | 描述 | | | |
| ADC0DIAG[1:0] = 0 | ADC0/ADC1转换为正常模式，且禁用诊断电流。 | ADC0/ADC1的预期差分结果。 | 短路。 | 无论PGA如何设置，主ADC读 ≈ 0 V。 |
| ADC0DIAG[1:0] = 1 | 通过将ADC0DIAG[1:0]置为1，可以启动ADC0上的50 μ A 诊断电流源。转换ADC0和ADC1。 | 主ADC电压变化为 $\Delta V = +50 \mu\text{A} \times R1$ 。例如，对于 $R1 = 2 \text{ k}\Omega$ ，约为100 mV | 在ADC0和ADC1之间形成短路在R1_a 和R1_b 之间形成短路。 | 无论PGA如何设置，主ADC读 ≈ 0 V。 |
| | 在单端模式下转换ADC0 (诊断电流已被禁用)。 | ADC0上的期望电压值 | ADC0开路或R1开路。 | 无论PGA如何设置，主ADC读 ≈ 0 V。 |
| ADC0DIAG[1:0] = 3 | 通过将ADC0DIAG[1:0]置为3，可以启动ADC0和ADC1上的50 μ A 诊断电流源。转换ADC0和ADC1。 | 主ADC的电压变化量为 $\Delta V = 50 \mu\text{A} \times (R1 - R2)$ ，即，约10 mV(公差: 10%)。 | R1与R2不匹配。 | 主ADC读 > 10 mV。 |

SINC3滤波器

通过向ADCFLT寄存器的[6:0]位写入数值，可以设置sinc3滤波器的抽取系数。表46和表47中对抽取系数值进行了详细的说明。

sinc3滤波器(SF)字的工作范围取决于斩波功能是否已启用。如果斩波功能被禁用，当ADC的频率在50 Hz到8 kHz范围内时，SF字的最小值为0，最大值为127。

有关如何基于ADCFLT寄存器SF[6:0]位的值计算ADC采样频率，见表46。更改转换速度时，重新启动之前应将ADC置于空闲模式。

ADC斩波

ADuC7060/ADuC7061内的ADC重复翻转其输入时，ADC可执行斩波操作。因此，从sinc3滤波器输出的抽取数字输出值会有与之相对应的正向、反向偏移量。ADC输出的这个结果包括一个最终的加法级，它可以将滤波器的当前输出值与前一个输出值进行求和及求平均值操作。之后，这个值将发送到ADC数据寄存器中。斩波可产生极好的直流偏移和偏移漂移性能，且非常适合于需要减少漂移和抑制噪声的应用场合。

可编程增益放大器

主ADC集成了一个片内可编程增益放大器(PGA)。PGA可设置为1至512范围内的10个不同数值。通过ADC0CON 寄存器的ADC0PGA[3:0]位，可控制增益。

激励源

ADuC7060/ADuC7061系列产品有两个配套的软件可编程电流源。这些激励电流来自AVDD。可对它们进行单独的配置，以产生不同大小的电流(200 μ A ~ 1 mA)。电流变化的步长为200 μ A。

这些电流源可以用来激励外部阻性电桥或RTD传感器。IEXCON 寄存器可控制激励电流源。IEXCON寄存器的位6必须置1，才可启用激励电流源0。同样，将IEXCON寄存器的位7置必须1，才可启用激励电流源1。该寄存器的IOUT[3:0]位用于控制每个电流源的输出电流。

此外，还可以通过设置IEXCON的IEXC0_DIR和IEXC1_DIR位，控制激励电流源向单个输出引脚(IEXC0或IEXC1)输出电流。这样，从单个激励引脚输出的电流的最大值可达2 mA。

ADC低功耗模式

ADuC7060/ADuC7061系列产品中的主ADC和辅助ADC均可在低功耗模式下工作。进入该模式，可减少ADC的吞吐时间，与正常模式相比，主ADC的功耗降低四分之一，辅助ADC的功耗降低约三分之一。在该模式下，最大的ADC转换速率为2 kHz。ADCMDE

寄存器用于控制ADC的工作模式。通过对该寄存器进行设置，可以让器件在正常模式(默认)、低功耗模式或者超低功耗模式下工作。超低功耗模式与低功耗模式的区别在于，在超低功耗模式状态下，PGA被禁用。按以下步骤操作，可让ADC进入低功耗模式：

- ADCMDE[4:3]--置1，ADC进入正常模式、低功耗模式或者超低功耗模式。
- ADCMDE[5]—置1，让器件进入低功耗模式。
- ADCMDE[7]-该位清0，可降低ADC时钟频率，从而降低器件的功耗。

ADC比较器和累加器

由主ADC产生的每个转换结果都可以和ADCCFG[4:3]的预设阈值(ADC0TH)进行比较。如转换结果的绝对值(符号无关)大于预设比较器阈值,就会产生MCU中断。作为比较器的扩展功能,可以利用用户代码配置一个阈值计数器(ADC0THV),以监控主ADC转换结果高于或低于预设阈值的次数。另外,当阈值计数器的值达到预设值(ADC0RCR)后,将产生ADC中断。

最后,还可通过控制ADCCFG[6:5]位来设置集成的32位累加器(ADC0ACC),从而允许主ADC将多个采样结果相加或相减。用户代码可直接读取累加值(ADC0ACC),而不需要任何进一步的软件处理。

温度传感器

ADuC7060/ADuC7061系列产品提供从片内带隙基准电压输出并与绝对温度成比例的电压。该电压还可以通过辅助ADC前端多路复用器有效地连接到ADC,构成一个附加的ADC输入通道,这样就可以很方便地形成一个内部温度传感器通道,用于测量芯片温度。

内部温度传感器的设计初衷不是作为绝对环境温度计算器。它用来作为ADuC7060/ADuC7061裸片温度的近似指示器。

典型温度系数为0.28 mV/°C。

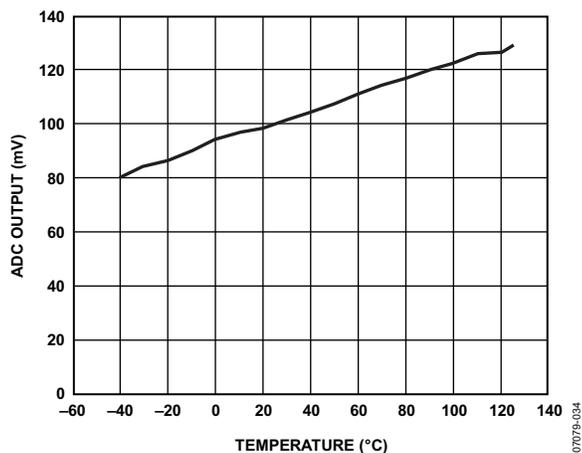


图16. ADC输出与温度的关系

ADC寄存器接口

ADC由一系列寄存器来控制 and 配置。各寄存器的详细描述见下文中的说明。

为了响应ADC中断,用户代码将会查询ADCSTA以确定中断源。通过设置ADCMSKI 寄存器,可单独屏蔽各个ADC中断源。关于ADCMSKI寄存器的说明,见表41。

主ADC的所有结果就绪位在ADC0DAT 寄存器读取后即被清0。如果主通道ADC处于禁用状态,那么所有ADC结果就绪位将在读取ADC1DAT 寄存器后被清0。为了确保主ADC和辅助ADC的输出数据是同步的,用户代码应先读取ADC1DAT 寄存器,再读取ADC0DAT 寄存器。只有先将对应的ADC结果就绪位清0,才能将新的ADC转换结果写入到ADCxDAT 寄存器中。这条规则的唯一例外情况是,数据转换结果正好在ARM内核断电时更新。在此模式下,即使没有将就绪位清0,ADCxDAT寄存器仍保存最近一次的ADC转换结果。

ADC状态寄存器

名称: ADCSTA
地址: 0xFFFF0500
默认值: 0x0000
访问类型: 只读
功能: 该只读寄存器用于保存与ADuC7060 ADuC7061 ADC工作模式或电流状态相关的一般状态信息。

表40 ADCSTA 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|-----------|---|
| 15 | ADCCALSTA | ADC校准状态。 硬件自动将该位置1，表示一个ADC校准周期已完成。 对ADCMDE进行写操作后，该位被清零。 |
| 14 | | 未使用。 该位保留供未来使用。 |
| 13 | ADC1CERR | 辅助ADC转换错误。 硬件自动将该位置1，表示辅助ADC转换超量程或欠量程。此时将转换结果箝位至负满量程(欠量程错误)或正满量程(超量程错误)。向ADC1DAT寄存器写入一个有效的(量程范围内的)电压转换结果后，该位被清0。 |
| 12 | ADC0CERR | 主ADC转换错误。 硬件自动将该位置1，表示主ADC转换超量程或欠量程。此时将转换结果箝位至负满量程(欠量程错误)或正满量程(超量程错误)。 向ADC0DAT寄存器写入一个有效的(量程范围内的)转换结果后，该位被清0。 |
| 11:7 | | 未使用。这些位保留供未来其他功能使用，不必由用户代码监控。 |
| 6 | ADC0ATHEX | ADC0累加器比较器阈值超出。 当保存在ADC0ACC中的ADC0累加器的值超出ADC0比较器阈值寄存器ADC0ATH中的阈值(通过编程设定此值)时，该位置1。 当ADC0ACC寄存器的值未超过ADC0ATH中的值时，该位清0。 |
| 5 | | 未使用。该位保留供未来其他功能使用，不必由用户代码监控。 |
| 4 | ADC0THEX | 主通道ADC比较器阈值。该位仅在当主通道ADC比较器由ADCCFG寄存器使能的条件下有效。 在主ADC转换结果绝对值大于ADC0TH寄存器写入值时，硬件将该位置1。如果使用ADC阈值计数器(ADC0RCR)，该位仅在指定的主ADC转换次数等于ADC0THV寄存器值时置1。 否则，该位清0。 |
| 3 | ADC0OVR | 主通道ADC超量程位。利用ADCCFG寄存器使能超量程检测功能后，如果主ADC输入明显超过量程(约大于30%)，硬件自动将该位置1。该位每隔125 us更新一次。该位置1后，仅当ADCCFG[2]清0以禁用此功能或通过ADC0CON寄存器修改ADC增益时，该位由软件清0。 |
| 2 | | 未使用。该位保留供未来其他功能使用，不必由用户代码监控。 |
| 1 | ADC1RDY | 辅助ADC结果就绪位。 当辅助通道ADC使能后，一旦有效转换结果写入ADC1DAT寄存器，该位即可通过软件置1。每次校准时序结束时，该位也会被置1。 在分别对ADC1DAT和ADC0DAT进行读操作后，该位清0。无论主ADC是否可用，都必须通过执行读ADC0DAT操作来清除该位。 |
| 0 | ADC0RDY | 主ADC结果就绪位。 当主通道ADC使能后，一旦有效转换结果写入ADC0DAT寄存器，该位即可通过软件置1。每次校准时序结束时，该位也会被置1。 执行读ADC0DAT操作，可清除该位。 |

ADuC7060/ADuC7061

ADC中断屏蔽寄存器

名称: ADCMSKI
地址: 0xFFFF0504
默认值: 0x0000
访问类型: 读、写
功能: 该寄存器可单独使能ADC中断源。此寄存器的位位置与ADCSTA寄存器的低八位相同。若一位由用户代码置1, 相应中断使能。默认情况下, 所有位为0, 表示禁用所有ADC中断源。

表41. ADCMSKI 寄存器位功能描述

| 位 | 名称 | 描述 |
|---|-----------------|---|
| 7 | | 未使用。该位保留供未来其他功能使用, 不必由用户代码监控。 |
| 6 | ADC0ATHEX_INTEN | ADC0累加器比较器阈值超出中断使能位。 当该位的值为1时, 若果ADCSTA寄存器的ADC0ATHEX位置1, 则使能产生中断。如果该位被清零, 则禁用该中断源。 |
| 5 | | 未使用。该位保留供未来其他功能使用, 不必由用户代码监控。 |
| 4 | ADC0THEX_INTEN | 主通道ADC比较器阈值超出中断使能位。 当该位的值为1时, 如果ADCSTA寄存器的ADC0THEX位置1, 则使能产生中断。 如果该位被清零, 则禁用该中断源。 |
| 3 | ADC0OVR_INTEN | 当该位的值为1时, 如果ADCSTA寄存器的ADC0OVR位置1, 则使能产生中断。如果该位被清零, 则禁用该中断源。 |
| 2 | | 未使用。该位保留供未来其他功能使用, 不必由用户代码监控。 |
| 1 | ADC1RDY_INTEN | 辅助ADC结果就绪位。 当该位的值为1时, 如果ADCSTA寄存器的ADC1RDY位置1, 则使能产生中断。如果该位被清零, 则禁用该中断源。 |
| 0 | ADC0RDY_INTEN | 主ADC结果就绪位。 当该位的值为1时, 如果ADCSTA寄存器的ADC0RDY位置1, 则使能产生中断。如果该位被清零, 则禁用该中断源。 |

ADC模式寄存器

名称: ADCMDE
地址: 0xFFFF0508
默认值: 0x03
访问类型: 读、写
功能: ADC模式寄存器是一个8位寄存器, 用于配置ADC子系统的工作模式。

表42. ADCMDE 寄存器位功能描述

| 位 | 名称 | 描述 |
|---|-----------|--|
| 7 | ADCCLKSEL | 将该位置1, 可将ADC时钟频率设置为512 kHz。当ADC处于正常工作模式下, 应将该位置1。将该位清0, 可将ADC时钟频率设置为131 kHz。当ADC处于低功耗工作模式下, 应将该位清0。 |
| 6 | | 未使用。该位保留供未来其他功能使用, 不必由用户代码监控。 |
| 5 | ADCLPMEN | 启用低功耗模式。当ADCMDE[4:3] = 00(ADC处于正常工作模式)时, 该位不起作用。 ADC工作在低功耗模式下时, 必须将该位置1。 ADC工作在低功耗模式下时, 如将该位清0, 将导致ADC输出结果不稳定。 |

| 位 | 名称 | 描述 |
|-----|----------------|---|
| 4:3 | ADCLPMCFG[1:0] | <p>ADC功耗模式配置。</p> <p>[00] = ADC正常模式。使能后，ADC以正常功耗工作，获得最佳电气性能。</p> <p>[01] = ADC低功耗模式。</p> <p>[10] = ADC正常模式，同[00]。</p> <p>[11] = ADC超低功耗模式(低功耗模式，且关断PGA)。</p> |
| 2:0 | ADCMD[2:0] | <p>ADC工作模式配置。</p> <p>[000] = ADC省电模式。所有ADC电路和输入放大器关断。</p> <p>[001] = ADC连续转换模式。在该模式下，任一被启用的ADC将连续以f_{ADC}这一频率进行转换。在向ADC0DAT/ADC1DAT写入新数据前，须将ADCxRDY清0。</p> <p>[010] = ADC单转换模式。在此模式下，任一被启用的ADC将执行一次转换。ADC在完成单次转换操作后进入空闲模式。一个单次转换需要花费两到三个ADC时钟周期，具体的周期数取决于是否启用了斩波模式。</p> <p>[011] = ADC空闲模式。在此模式下，ADC完全上电，但仍保持复位状态。校准操作完成后，器件进入该模式。</p> <p>[100] = ADC自失调校准。在该模式下，器件借助一个内部产生的0 V电压对任何使能的ADC进行失调校准操作。校准以用户编程的ADC设置作为依据；因此，正常的单一ADC转换在校准结果产生前需要花费两到三个ADC转换周期。校准结果自动写入相应ADC的ADCxOF寄存器内。失调校准周期结束后，ADC返回空闲模式，且校准和转换就绪状态位置1。</p> <p>注意：应利用ADC0提供主ADC上的单端自校准周期。应利用ADC0/ADC1对主ADC的差分输入进行自校准。</p> <p>[101] = ADC自增益校准。在此模式下，参照内部基准电压校准所有使能后ADC的增益。增益校准分为两个阶段，耗时是失调校准的两倍。校准结果自动写入对应ADC的ADCxGN寄存器。增益校准周期结束后，ADC返回空闲模式，校准和转换就绪状态位置1。只有主通道ADC才能进行ADC自增益校准。</p> <p>注意：仅在增益为1时，自增益校准功能才可用；增益大于1时，该功能不可用。</p> <p>[110] = ADC系统零电平校准。在该模式下，参照ADC输入引脚处驱动的外部零电平电压，对使能ADC通道进行零电平校准。这样，可从外部实现通道短路。</p> <p>[111] = ADC系统满量程校准。在此模式下，参照ADC输入引脚处驱动的外部满量程电压对使能ADC通道进行满量程校准。在满量程校准时序执行完毕后，ADCxGN寄存器被更新。</p> |

主ADC控制寄存器

| | |
|-------|--|
| 名称： | ADC0CON |
| 地址： | 0xFFFF050C |
| 默认值： | 0x8000 |
| 访问类型： | 读、写 |
| 功能： | 主通道ADC控制寄存器是一个16位寄存器。通过ADC0CON对主ADC进行重配置后，辅助ADC也将复位。 |

ADuC7060/ADuC7061

表43. ADC0CON 寄存器位功能描述及位名称

| 位 | 名称 | 描述 |
|-------|---------------|--|
| 15 | ADC0EN | 主通道ADC使能。 由用户代码将该位置1，使能主ADC。 将该位清0，可关断主ADC，可导致ADCSTA 寄存器中的对应ADC就绪位复位至0。 |
| 14:13 | ADC0DIAG[1:0] | 诊断电流源使能位。 [00] = 关闭电流源。 [01] = 在所选正极输入端(例如: ADC0)，使能一个大小为50 μA的电流源。 [10] = 在所选负极输入端(例如: ADC1)，使能一个大小为50 μA的电流源。 [11] = 在所选的两个输入端(例如: ADC0和ADC1)，使能一个50 μA的电流源。 |
| 12 | HIGHEXTREF0 | 当ADC0的外部基准电压高于1.35 V时，必须将该位置1。这样，可以将基准电压源二分频。 当使用内部基准电压或外部基准电压低于1.35 V时，须将该位清0。 |
| 11 | AMP_CM | 用户将该位置1，可将PGA输出共模电压设置为AVDD/2。 通过用户代码将该位清0，可将PGA输出共模电压设置为PGA输入共模电压电平。 |
| 10 | ADC0CODE | 主通道ADC输出编码。 通过用户代码将该位置1，可将主ADC输出编码配置为单极性。 通过用户代码将该位清0，可将主ADC输出编码配置为二进制补码。 |
| 9:6 | ADC0CH[3:0] | 主通道ADC输入选择。注意：为ADC5选择单端通道。 ADC5偏置至0.1 V(最低电压值)。 [0000]= ADC0/ADC1 (差分模式) [0001] = ADC0/ADC5 (单端模式)。 [0010] = ADC1/ADC5 (单端模式)。 [0011] = VREF+、VREF-。注意：这是通过ADC0REF位选择的基准电压。 [0100] = 未使用。该位保留供未来其他功能使用，不应写入。 [0101] = ADC2/ADC3 (差分模式)。 [0110] = ADC2/ADC5 (单端模式)。 [0111] = ADC3/ADC5 (单端模式)。 [1000] = 内部短路至ADC1。 [1001] = 内部短路至ADC1。 |
| 5:4 | ADC0REF[1:0] | 选择主通道ADC基准电压。 [00] = 选择内部基准电压。当ADC处于低功耗模式下时，由ADCMDE[5]控制基准电压的选择。 [01] = 选择外部基准电压输入(VREF+/VREF-)。当基准电压高于1.3 V时，HIGHEXTREF0位置1。 [10] = 选择辅助外部基准电压输入(ADC4/EXT_REF2IN+、ADC5/EXT_REF2IN-)当基准电压高于1.3 V时，HIGHEXTREF0位置1。 [11] = 选择AVDD、AGND二分频。 |
| 3:0 | ADC0PGA[3:0]. | 选择主通道ADC增益。注意，标称主ADC满量程输入电压=(VREF/增益)。 [0000] = ADC0增益为1。负输入端缓冲器旁路。 [0001] = ADC0增益为2。 [0010] = ADC0增益为4(默认值)。使能仪表放大器。 [0011] = ADC0增益为8。 [0100] = ADC0增益为16。 [0101] = ADC0增益为32。 [0110] = ADC0增益为64(最大的PGA增益设置)。 [0111] = ADC0增益为128(额外的数字增益)。 [1000] = ADC0增益为256。 [1001] = ADC0增益为512。 [1XXX] = 未定义ADC0增益。 |

辅助ADC控制寄存器

名称: ADC1CON
 地址: 0xFFFFF0510
 默认值: 0x0000
 访问类型: 读、写
 功能: 辅助ADC控制寄存器是一个16位寄存器。

表44. ADC1CON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|---------------|---|
| 15 | ADC1EN | 使能辅助通道ADC。 由用户代码将该位置1, 将使能辅助ADC。 将该位清0, 可关断辅助ADC。 |
| 14:13 | ADC1DIAG[1:0] | 诊断电流源使能位。与ADC0DIAG[1:0]所用的为同一电流源。多个ADC无法同时使能诊断电流源。 [00] = 关闭电流源。 [01] = 在所选正极输入端(例如: ADC2)使能一个大小为50 μ A的电流源。 [10] = 在所选负极输入端(例如: ADC3)使能一个大小为50 μ A的电流源。 [11] = 在所选的两个输入端(例如: ADC2和ADC3)使能一个50 μ A的电流源。 |
| 12 | HIGHEXTREF1 | 当ADC1的外部基准电压高于1.35 V时, 必须将该位置1。这样, 可以将基准电压源二分频。 当使用内部基准电压或外部基准电压低于1.35 V时, 须将该位清0。 |
| 11 | ADC1CODE | 辅助通道ADC输出编码。 通过用户代码将该位置1, 可将辅助ADC输出编码配置为单极性。 通过用户代码将该位清0, 可将辅助ADC输出编码配置为二进制补码。 |
| 10:7 | ADC1CH[3:0] | 辅助通道ADC输入选择。注意: 为ADC5选择单端通道。 ADC5偏置至0.1 V(最低电压值)。 [0000] = ADC2/ADC3 (差分模式)。 [0001] = ADC4/ADC5 (差分模式)。 [0010] = ADC6/ADC7 (差分模式)。 [0011] = ADC8/ADC9 (差分模式)。 [0100] = ADC2/ADC5 (单端模式)。 [0101] = ADC3/ADC5 (单端模式)。 [0110] = ADC4/ADC5 (单端模式)。 [0111] = ADC6/ADC5 (单端模式)。 [1000] = ADC7/ADC5 (单端模式)。 [1001] = ADC8/ADC5 (单端模式)。 [1010] = ADC9/ADC5 (单端模式)。 [1011] = 内部温度传感器+/内部温度传感器 [1100] = VREF+, VREF-注意: 这是通过ADC1REF位选择的基准电压。 [1101] = DAC_OUT/AGND [1110] = 未定义。 [1111] = 内部短路至ADC3。 |

ADuC7060/ADuC7061

| 位 | 名称 | 描述 |
|-----|-----------------|--|
| 6:4 | ADC1REF[2:0] | 选择辅助通道ADC基准电压。 [000] = 选择内部基准电压。当ADC处于低功耗模式下时，由ADCMODE[5]控制基准电压的选择。 [001] = 选择外部基准电压输入 (VREF+/VREF-)。当基准电压高于1.3 V时，HIGHEXTREF1位置1。 [010] = 选择辅助外部基准电压输入(ADC4/EXT_REF2IN+、ADC5/EXT_REF2IN-)当基准电压高于1.35 V时，HIGHEXTREF1位置1。 [011] = 选择AVDD、AGND二分频。如果选择此配置，HIGHEXTREF1位自动置位1。 [100] = (AVDD, ADC3)ADC3可以作为基准电压源的负输入端。 [101] 至 [111] = 保留。 |
| 3:2 | BUF_BYPASS[1:0] | 缓冲器旁路。 [00] = 开启整个缓冲器。正负极缓冲器输入端均处于活动状态。 [01] = 负极缓冲器旁路，正极缓冲器开启。 [10] = 负极缓冲器开启，正极缓冲器旁路。 [11] = 整个缓冲器旁路。正负极缓冲器输入端均处于关闭状态。 |
| 1:0 | | 数字增益。为辅助ADC输入端选择数字增益。 [00] = ADC1增益为1。 [01] = ADC1增益为2。 [10] = ADC1增益为4。 [11] = ADC1增益为8。 |

ADC滤波器寄存器

名称: ADCFLT

地址: 0xFFFF0408

默认值: 0x0007

访问类型: 读、写

功能: ADC滤波器寄存器是一个16位寄存器，用于控制两个片内ADC的速度和分辨率。注意，修改ADCFLT将导致主ADC和辅助ADC复位。
更改转换速度时，重新启动之前应将ADC置于空闲模式。

表45. ADCFLT 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|---------|---|
| 15 | CHOPEN | 斩波使能。用户置1，可使能所有活动ADC的系统斩波。该位置1后，ADC失调误差和漂移都很低，但在AF = 0时，ADC输出速率将降至原来的三分之一(见此表中的Sinc3抽取系数位[6:0])。AF > 0时，输出更新速率在斩波开启或关闭时均相同。斩波使能时，建立时间为两个输出周期。 |
| 14 | RAVG2 | 除二移动平均功能使能位。 使能除二移动平均功能，可降低ADC噪声。当激活斩波功能后，该功能将自动使能。斩波不激活时可选，且使能后(斩波不激活时)不会降低ADC输出速率，而将建立时间增加一个转换周期。 用户可将该位清0，以禁用移动平均功能。 |
| 13:8 | AF[5:0] | 平均系数(AF)。写入这些位的数值用来实现可编程一阶Sinc3后滤波器。平均系数可进一步降低ADC噪声，但同时，ADC的输出速率(见本表中对Sinc3抽取系数位[6:0]的说明)也会降低。 |

| 位 | 名称 | 描述 |
|-----|---------|--|
| 7 | NOTCH2 | Sinc3修正。用户将该位置1,可修正标准Sinc3频率响应,以便将滤波器阻带抑制增加约5 dB。这可以通过插入另一个陷波(NOTCH2)来实现。该陷波的频率公式如下: $f_{NOTCH2} = 1.333 \times f_{NOTCH}$ 其中, f_{NOTCH} 表示响应中第一个陷波所在位置。 |
| 6:0 | SF[6:0] | Sinc3抽取系数(SF)1。写入这些位的值(SF)可控制Sinc3滤波器的过采样(抽取系数)。以下公式用于计算Sinc3滤波器的输出速度: $f_{ADC} = (512,000 / ([SF + 1] \times 64)) \text{ Hz}^2$ 当斩波位(位15,斩波使能)的值为0且平均系数(AF)的值也为0时,上述公式可用。这一公式适用于所有小于等于125的SF值。 SF = 126时,强制 f_{ADC} 为60 Hz。 SF = 127时,强制 f_{ADC} 为50 Hz。 有关SF(不包括126和127)和AF值的 f_{ADC} 计算信息,见表46。 |

¹ 由于受数字滤波器内部数据路径的限制,用来产生所需ADC输出速率的Sinc3抽取系数(SF)和平均系数(AF)的组合也有限制。此约束在正常功耗模式下将最低ADC更新速率限制为4 Hz,在更低功耗模式下将其限制为1 Hz。

² 在低功耗模式下,ADC由低功耗振荡器(131 kHz)直接驱动,而不是由512 kHz时钟驱动。计算 f_{ADC} 时,应(约)除以4。

表46. ADC转换速率和建立时间

| 斩波使能 | 平均系数 | 移动平均 | f_{ADC} 正常模式 | f_{ADC} 低功耗模式 | $t_{SETTLING}^1$ |
|------|------|------|--|--|---------------------|
| 无 | 无 | 无 | $\frac{512,000}{[SF + 1] \times 64}$ | $\frac{131,072}{[SF + 1] \times 64}$ | $\frac{3}{f_{ADC}}$ |
| 无 | 无 | 有 | $\frac{512,000}{[SF + 1] \times 64}$ | $\frac{131,072}{[SF + 1] \times 64}$ | $\frac{4}{f_{ADC}}$ |
| 无 | 有 | 无 | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF]}$ | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF]}$ | $\frac{1}{f_{ADC}}$ |
| 无 | 有 | 有 | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF]}$ | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF]}$ | $\frac{2}{f_{ADC}}$ |
| 有 | 不可用 | 不可用 | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF] + 3}$ | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF] + 3}$ | $\frac{2}{f_{ADC}}$ |

¹ 在进行第一次模数转换前,每个ADC需要约60 μ s的额外时间。

表47. SF和AF的允许组合

| SF | AF范围 | | |
|--------|------|-----|------|
| | 0 | 1至7 | 8至63 |
| 0至31 | 有 | 有 | 有 |
| 32至63 | 有 | 有 | 无 |
| 64至127 | 有 | 无 | 无 |

ADuC7060/ADuC7061

ADC配置寄存器

| | |
|-------|-------------------------------|
| 名称: | ADCCFG |
| 地址: | 0xFFFF0518 |
| 默认值: | 0x00 |
| 访问类型: | 读、写 |
| 功能: | 该8位ADC配置寄存器用于控制与片内ADC相关的扩展功能。 |

表48. ADCCFG 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|----------------|---|
| 7 | GND_SW_EN | 模拟接地开关使能。 用户软件将该位置1, 可将外部GND_SW引脚连接到内部模拟地基准点。在程序控制下, 该位可用于将外部电路和部件接地或断开, 从而在不使用外部电路或部件时, 将直流功耗降到最低。该位与ADCCFG[1]配合, 可选择20 kΩ电阻接地。 若将该位清零, 可断开模拟地开关与外部引脚的连接。 |
| 6:5 | ADC0ACCEN[1:0] | 主通道(32位)累加器使能。 [00] = 禁用累加器, 再将其复位至0。一次完整的模数转换(ADCSTA[0]设置两次)完成后, 必须禁用累加器, 然后再使能累加器, 以确保将其复位。 [01] = 激活累加器。正电流值相加到累加器总数之中; 如果转换次数超过65,535次, 累加器将产生溢出。负电流值从累加器总数中减去; 累加器箝位到最小值0。 [10] = 激活累加器。除不产生箝位操作外, 其他功能与[01]的相同。正电流值相加到累加器总数之中; 如果转换次数超过65,535次, 累加器将产生溢出。负电流的绝对值从累加器总数中减去; 在此模式下, 累加器继续进行负向累加, 其值小于0。 [11] = 激活累加器和比较器。如果ADCMSK[6]置1, 该操作将引起ADC0中断。 |
| 4:3 | ADC0CMPEN[1:0] | 主ADC比较器使能位。 [00] = 禁用比较器。 [01] = 激活比较器。当ADC0转换结果的绝对值 $ \geq \text{ADC0TH}$ 时, 产生中断断言。 [10] = 激活比较器计数模式。当ADC0转换结果的绝对值 $ \geq \text{ADC0TH}$ (ADC0THC转换的数值)时, 产生中断断言。若转换结果 $ < \text{ADC0TH}$, 则阈值计数器的值(ADC0THV)复位为0。 [11] = 激活比较器计数模式。当ADC0转换结果的绝对值 $ \geq \text{ADC0TH}$ (ADC0THC转换的数值)时, 产生中断断言。若转换结果 $ < \text{ADC0TH}$, 则产生计数器阈值(ADC0THV)向0递减。 |
| 2 | ADC0OREN | ADC0超量程使能位。 用户将该位置1, 可以在主通道ADC上使能粗调比较器。如果读数明显超出有效增益设置(约>30%), 则ADCSTA寄存器的超量程位将置1。ADC读数必须超出该范围且在125 μs或更长的时间内维持该状态, 以便标志位置1。 当ADC处于低功耗模式时, 不要使用该特性。 |
| 1 | GND_SW_RES_EN | 将该位置1, 可使能一个与接地开关串联的20 kΩ电阻。 将该位清0, 可禁用上述电阻。 |
| 0 | ADCRNEN | ADC结果计数器使能位。 用户将该位置1, 可使能结果计数模式。当ADCORCR = ADCORCV时, 产生ADC中断。 将该位清0, 可禁用结果计数器。当每一次转换完成后, 都会产生ADC中断。 |

主通道ADC数据寄存器

名称： ADC0DAT
 地址： 0xFFFF051C
 默认值： 0x00000000
 访问类型： 只读
 功能： 该ADC数据MMR用于保存主ADC输出的24位转换结果。如果ADC0转换结果就绪位(ADCSTA[0])置1，则ADC不更新此MMR。MCU读取该寄存器后，将清除所有声明的就绪标志(ADCSTA[1:0])。

表49. ADC0DAT 寄存器位功能描述

| 位 | 描述 |
|------|---------------|
| 23:0 | ADC0 24位转换结果。 |

辅助通道ADC数据寄存器

名称： ADC1DAT
 地址： 0xFFFF0520
 默认值： 0x00000000
 访问类型： 只读
 功能： 该ADC数据寄存器用于保存辅助ADC输出的24位转换结果。如ADC0转换结果就绪位(ADCSTA[1])置1，则ADC无法更新此寄存器。

表50. ADC1DAT 寄存器位功能描述

| 位 | 描述 |
|------|---------------|
| 23:0 | ADC1 24位转换结果。 |

主通道ADC失调校准寄存器

名称： ADC0OF
 地址： 0xFFFF0524
 默认值： 特定器件，工厂编程
 访问类型： 读、写
 功能： 该ADC失调寄存器用于保存主ADC的16位失调校准系数。上电时该寄存器配置为出厂默认值。然而，如果用户通过ADCMDE寄存器的相应位对主ADC进行失调校准，则该寄存器自动覆写。只有在ADC处于空闲模式时，才允许用户代码对该校准寄存器进行写操作。在对任何失调寄存器或增益寄存器进行写操作前，必须将ADC使能且使其处于空闲模式。空闲模式必须持续至少23 μ s。

表51. ADC0OF 寄存器位功能描述

| 位 | 描述 |
|------|----------------|
| 15:0 | ADC0 16位失调校准值。 |

辅助通道ADC失调校准寄存器

名称： ADC1OF
 地址： 0xFFFF0528
 默认值： 特定器件，工厂编程
 访问类型： 读、写
 功能： 该失调寄存器用于保存辅助通道的16位失调校准系数。上电时该寄存器配置为出厂默认值。但是，如果用户利用ADCMDE寄存器的相应位对辅助通道进行失调校准，寄存器自动覆写。只有在ADC处于空闲模式时，才允许用户代码对该校准寄存器进行写操作。在对任何失调寄存器或增益寄存器进行写操作前，必须将ADC使能且使其处于空闲模式。空闲模式必须持续至少23 μ s。

表52. ADC1OF 寄存器位功能描述

| 位 | 描述 |
|------|----------------|
| 15:0 | ADC1 16位失调校准值。 |

ADuC7060/ADuC7061

主通道ADC增益校准寄存器

名称: ADC0GN
地址: 0xFFFF052C
默认值: 特定器件, 工厂编程
访问类型: 读、写
功能: 该增益寄存器可保存用于按比例调整主ADC转换结果的16位增益校准系数。上电时该寄存器配置为出厂默认值。然而, 如果用户利用ADCMDE寄存器的对应位对主ADC进行增益校准, 寄存器自动覆写。只有在ADC处于空闲模式时, 才允许用户代码对该校准寄存器进行写操作。在对任何失调寄存器或增益寄存器进行写操作前, 必须将ADC使能且使其处于空闲模式。空闲模式必须持续至少23 μ s。

表53. ADC0GN 寄存器位功能描述

| 位 | 描述 |
|------|----------------|
| 15:0 | ADC0 16位校准增益值。 |

辅助通道增益校准寄存器

名称: ADC1GN
地址: 0xFFFF0530
默认值: 特定器件, 工厂编程
访问类型: 读、写
功能: 该增益寄存器可保存用于按比例调整辅助通道转换结果的16位增益校准系数。上电时该寄存器配置为出厂默认值。

但是, 如果用户利用ADCMDE寄存器的相应位对辅助通道进行增益校准, 寄存器自动覆写。只有在ADC处于空闲模式时, 才允许用户代码对该校准寄存器进行写操作。在对任何失调寄存器或增益寄存器进行写操作前, 必须将ADC使能且使其处于空闲模式。空闲模式必须持续至少23 μ s。

表54. ADC1GN 寄存器位功能描述

| 位 | 描述 |
|------|----------------|
| 15:0 | ADC1 16位增益校准值。 |

主通道ADC结果计数器限值寄存器

名称: ADC0RCR
地址: 0xFFFF0534
默认值: 0x0001
访问类型: 读、写
功能: 该16位寄存器用于设置产生ADC中断前所必需的转换次数。默认情况下, 该寄存器的值为0x01。将DCCFG寄存器中的ADC结果计数器使能位置位, 可使能ADC计数器功能。

表55. ADC0RCR 寄存器位功能描述

| 位 | 描述 |
|------|--------------------|
| 15:0 | ADC0结果计数器限值/重载寄存器。 |

主通道ADC结果计数器寄存器

名称: ADC0RCV
地址: 0xFFFF0538
默认值: 0x0000
访问类型: 只读
功能: 该16位只读寄存器用于保存主ADC的当前转换结果。它和ADC0RCR配合使用, 可屏蔽主通道ADC中断, 从而产生较低的中断速率。当ADC0RCV = ADC0RCL时, ADC0RCV值复位至0, 且重新开始计数。它还可与累加器(ADC0ACC)配合使用, 进行平均值的计算。通过设置ADCCFG[0], 可以使能结果计数器。当重新配置主ADC(即当对ADC0CON或ADCMDE进行写操作)后, 该寄存器复位至0。

表56. ADC0RCV 寄存器位功能描述

| 位 | 描述 |
|------|---------------|
| 15:0 | ADC0结果计数器寄存器。 |

主通道ADC阈值寄存器

名称： ADC0TH
 地址： 0xFFFF053C
 默认值： 0x0000
 访问类型： 读、写
 功能： 该16位寄存器可设置用来与主ADC转换结果的绝对值进行比较的阈值。在单极性模式下，与ADC0TH[15:0]进行比较；而在二进制补码模式下，与ADC0TH[14:0]进行比较。

表57. ADC0TH 寄存器位功能描述

| 位 | 描述 |
|------|-------------------|
| 15:0 | ADC0 16位比较器阈值寄存器。 |

主通道ADC阈值计数器限值寄存器

名称： ADC0THC
 地址： 0xFFFF0540
 默认值： 0x0001
 访问类型： 读、写
 功能： 该8位寄存器用于决定在设置ADCSTA 寄存器的主ADC比较器阈值位，产生中断前ADC转换结果读数超过ADC0TH必须达到的累积次数（即低于递减阈值或将计数复位至0的值）。只要ADC0THV = ADC0RCR时，主ADC比较器阈值位立即置位。

表58. ADC0THC 寄存器位功能描述

| 位 | 描述 |
|------|--------------------|
| 15:8 | 保留。 |
| 7:0 | ADC0 8位阈值计数器限值寄存器。 |

主通道ADC阈值计数器寄存器

名称： ADC0THV
 地址： 0xFFFF0544
 默认值： 0x0000
 访问类型： 只读
 功能： 每当主ADC转换结果绝对值 $|Result| \geq ADC0TH$ ，该8位寄存器的值递增一次。每当主ADC转换结果 $|Result| < ADC0TH$ 时，该寄存器的值递减一次或复位至0。通过设置ADCCFG 寄存器的主通道ADC比较器位，可对该功能进行配置。

表59. ADC0THV 寄存器位功能描述

| 位 | 描述 |
|-----|--------------------|
| 7:0 | ADC0 8位阈值超过计数器寄存器。 |

主通道ADC累加器寄存器

名称： ADC0ACC
 地址： 0xFFFF0548
 默认值： 0x00000000
 访问类型： 只读
 功能： 该32位寄存器用于保存主ADC累加器的值。ADCSTA 寄存器的主ADC就绪位的值决定了何时可以安全地读取该寄存器。在ADCCFG 寄存器内禁用累加器或重新配置主通道ADC，可将寄存器的值复位至0。

表60. ADC0ACC 寄存器位功能描述

| 位 | 描述 |
|------|-----------------|
| 31:0 | ADC0 32位累加器寄存器。 |

ADuC7060/ADuC7061

主通道ADC阈值寄存器

名称: ADC0ATH
 地址: 0xFFFF054C
 默认值: 0x00000000
 访问类型: 读、写
 功能: 该32位寄存器用于保存主通道累加器比较器的阈值。当ADC0ACC中的累加器的值大于ADC0ATH中的值时, ADCSTA中的ADC0ATHEX位被置位。在此情况下, 如果ADCMSKI的相应位也使能, 则将产生中断。

表61. ADC0ATH 寄存器位功能描述

| 位 | 描述 |
|------|----------------------|
| 31:0 | ADC0 32位累加器比较器阈值寄存器。 |

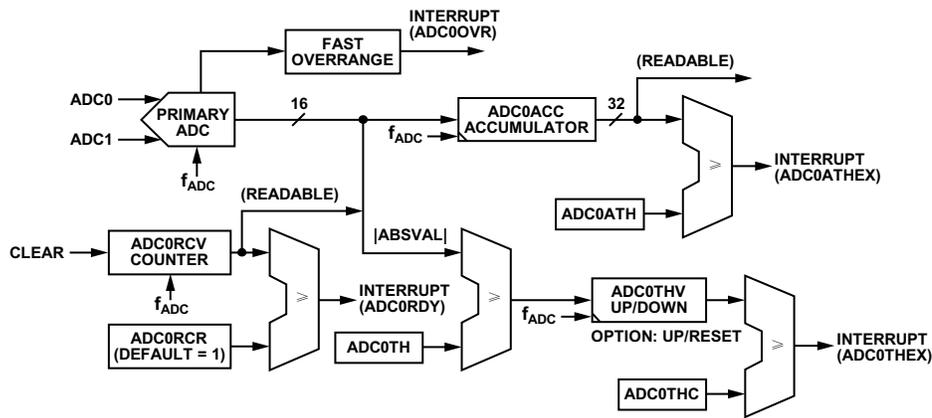


图17. 主ADC累加器/比较器/计数器模块框图

激励电流源控制寄存器

| | |
|-------|-------------------------------|
| 名称: | IEXCON |
| 地址: | 0xFFFF0570 |
| 默认值: | 0x00 |
| 访问类型: | 读、写 |
| 功能: | 该8位寄存器用于控制两个激励电流源IEXC0和IEXC1。 |

表62. IEXCON寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|-----------|---|
| 7 | IEXC1_EN | IEXC1电流源使能位。 将该位置1, 可以使能激励电流源1。 将该位清0, 可以禁用激励电流源1。 |
| 6 | IEXC0_EN | IEXC0电流源使能位。 将该位置1, 可以使能激励电流源0。 将该位清0, 可以禁用激励电流源0。 |
| 5 | IEXC1_DIR | 将该位置1, 可以将激励电流源1产生的激励电流送到IEXC0引脚。 将该位清0, 可以将激励电流源1产生的激励电流送到IEXC1引脚。 |
| 4 | IEXC0_DIR | 将该位置1, 可以将激励电流源0产生的激励电流送到IEXC1引脚。 将该位置0, 可以将激励电流源0产生的激励电流送到IEXC0引脚。 |
| 3:1 | IOUT[3:1] | 这些位用于控制每个激励电流源的激励电流电平。 IOUT[3:1] = 000, 激励电流 = $0 \mu\text{A} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 IOUT[3:1] = 001, 激励电流 = $200 \mu\text{A} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 IOUT[3:1] = 010, 激励电流 = $400 \mu\text{A} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 IOUT[3:1] = 011, 激励电流 = $600 \mu\text{A} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 IOUT[3:1] = 100, 激励电流 = $800 \mu\text{A} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 IOUT[3:1] = 101, 激励电流 = $1 \text{ mA} + (\text{IOUT}[0] \times 10 \mu\text{A})$ 。 其它所有值均未定义。 |
| 0 | IOUT[0] | 将该位置1, 可以使能10 μA 诊断电流源。 将该位清0, 可以禁用10 μA 诊断电流源。 |

应用电路示例

图18显示了ADuC7060/ADuC7061上的一个简单的桥式传感器接口, 包括模拟输入通道上的RC滤波器。注意, 桥式传感器上的传感线(与基准输入端相连)分别与不同的激励线(接至DVDD/AVDD和地)相连。因此, 共有六根线与桥式传感器相连。六线连接是大多数现有桥式传感器(如称重传感器)的特性之一, 借助该特性, 可将由线阻抗引起的误差降至最低。

在图19中, AD592是一个外部温度传感器, 用于测量热电偶的冷端, 其输出端与辅助通道相连。ADR280是一个外部1.2 V基准器件; 也可用内部基准电压源来代替它。热电偶与主ADC相连, 为ADC0/ADC1提供差分输入。注意: 位于VREF+和ADC1之间的电阻可将ADC输入信号偏置100 mV以上。

图20显示了一个简单的四线RTD接口电路。图18中显示了一个桥式传感器, 在此基础上, 如再连接一个电源且通过串口与一外部器件相连, 则构成一个完整的系统(如图20)。

ADuC7060/ADuC7061

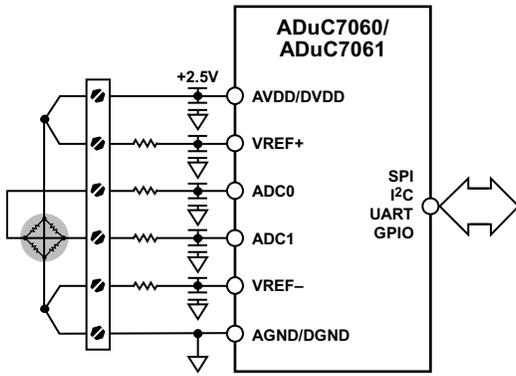


图18. 桥式接口电路

07079-012

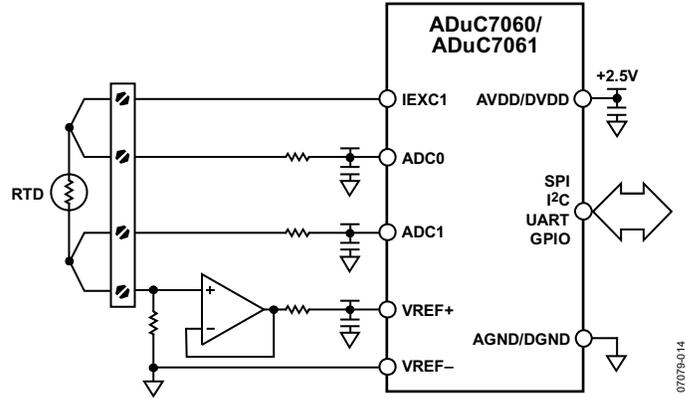


图20. RTD接口电路示例

07079-014

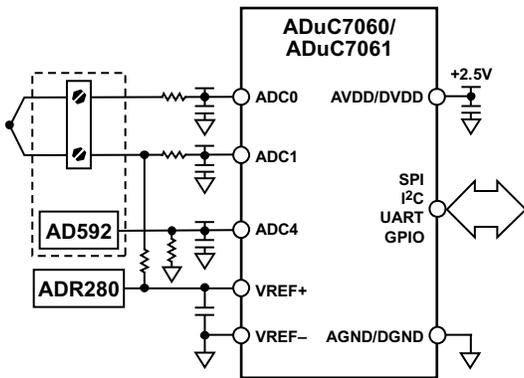


图19. 热电偶接口电路示例

07079-013

DAC外设

DAC

ADuC7060/ADuC7061集成了一个片内电压输出DAC。在正常模式下，DAC的分辨率为12位。在插值模式下，DAC的分辨率为16位(14个有效位)。DAC有一个轨到轨电压输出缓冲器(驱动能力为5 kΩ/100 pF)。

DAC的4个可选电压范围包括：

- 0 V至 V_{REF} (内部带隙1.2 V基准电压)
- V_{REF-} 至 V_{REF+}
- ADC5/EXT_REF2IN-至ADC4/EXT_REF2IN+
- 0 V至AVDD

最大信号范围为0 V至AVDD。

运算放大器模式

作为一个选项，可禁用DAC，它的输出缓冲器可以起到运算放大器的作用。

寄存器接口

DAC可通过一个控制寄存器和一个数据寄存器进行配置。

DAC0CON寄存器

| | |
|-------|------------|
| 名称： | DAC0CON |
| 地址： | 0xFFFF0600 |
| 默认值： | 0x0200 |
| 访问类型： | 读、写 |

表63. DAC0CON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------------|---|
| 15:10 | | 保留。 |
| 9 | DACPD | 将该位置1，可以关断DAC输出(DAC输出为三态)。将该位清0，可以使能DAC。 |
| 8 | DACBUFLP | 将该位置1，可以让DAC输出缓冲器在低功耗模式下工作。关于电气规格的详细说明，请参阅“正常DAC模式与运算放大器模式”部分。将该位清0，可以使能DAC缓冲器。 |
| 7 | OPAMP | 将该位置1，可以让DAC输出缓冲器在运算放大器模式下工作。将该位清0，可以使能DAC输出缓冲器用于正常DAC操作。 |
| 6 | DACBUFBYPASS | 将该位置1，可以将输出缓冲器旁路，并直接向输出引脚输出DAC转换结果。如将该位清0，DAC转换结果将输出到缓冲器中。 |
| 5 | DACCLK | 将该位清0，以在HCLK负边沿更新DAC。将该位置1，以在定时器0负边沿更新DAC。对波形生成来说，这是一个非常理想的模式。在这种模式下，波形的下一个值可在正常的定时器0时间间隔后被写入DAC0DAT。 |
| 4 | DACCLR | 将该位置1，以便进行正常的DAC操作。将该位清0，可清除DAC输出结果并向DAC0DAT内写入0。向该位写入数值，会立即对DAC输出结果产生影响。 |
| 3 | DACMODE | 将该位置1，可以让DAC在16位插值模式下工作。将该位清0，可以让DAC在12位正常模式下工作。 |
| 2 | Rate | 用于DAC的插值模式。将该位置1，可将插值时钟设置为UCLK/16。 |
| 1:0 | DAC range bits | [11] = 0 V至AVDD。 [10] = ADC5/EXT_REF2IN-至ADC4/EXT_REF2IN+ [01] = V_{REF-} 至 V_{REF+} [00] = 0 V至 V_{REF} (1.2 V)内部基准电压源。 |

ADuC7060/ADuC7061

DAC0DAT寄存器

名称: DAC0DAT
地址: 0xFFFF0604
默认值: 0x00000000
访问类型: 读、写
功能: 该32位寄存器包含DAC输出值。

表64.DAC0DAT 寄存器位功能描述

| 位 | 描述 |
|-------|-----------------|
| 31:28 | 保留。 |
| 27:16 | DAC0的12位数据。 |
| 15:12 | 额外的4位, 可用于插值模式。 |
| 11:0 | 保留。 |

DAC的用法

片内DAC架构由一电阻串DAC和一个输出缓冲放大器构成。

用户可通过软件来选择DAC基准电压源。AVDD、VREF±或者ADCx/EXT_REF2IN±均可作为DAC基准电压源。

- 在0至AVDD模式下, DAC输出传递函数在AVDD引脚上的范围为0 V至AVDD;
- 在VREF±和ADCx/EXT_REF2IN±模式下, DAC输出传递函数为负输入引脚电压至正输入引脚电压。注意, 上述电压不得低于0 V或高于AVDD。
- 在0至VREF模式时, DAC输出传递函数为0至1.2 V内部基准电压源VREF。

DAC可被配置为以下三种不同的用户模式: 正常模式、DAC插值模式和运算放大器模式。

正常DAC模式

在该模式下, DAC配置为12位电压输出DAC。在默认的情况下, 可以使用DAC缓冲器, 但输出缓冲器可以被禁用。当DAC输出缓冲器禁用时, DAC仅可驱动20 pF的容性负载。通过对DAC0CON寄存器的DACBUFBYPASS位进行设置, 可以禁用DAC缓冲器。

每一个DAC输出缓冲放大器都有一个真轨到轨输出级。也就是说, 当输出空载时, DAC输出摆幅能够达到AVDD或接地电平的5 mV范围以内。此外, 当驱动一个5 kΩ电阻负载到地时, 除了代码0至100(在0至AVDD模式下时代码3995至4095), 整个传递函数都能保证符合DAC线性度规格要求。

地附近和AVDD处的线性降级是由输出放大器的饱和引起的, 图21反映了这种效应的一般表现(失调误差和增益误差忽略不计)。图21中的虚线为理想的传递函数, 实线代表转换可能具有端点非线性(由输出放大器饱和引起)的传递函数。注意, 图21仅代表输入范围为0至AVDD模式时的传递函数。在0至VREF模式、VREF±或ADCx/EXT_REF2IN±模式下, 低端具有相似的非线性; 这时, 通常VREF < AVDD或ADCx/EXT_REF2IN± < AVDD。然而, 传递函数的上半部分一直到端点都表现为理想的线性(这里采用VREF, 而非AVDD)。这说明DAC输出没有端点线性误差。

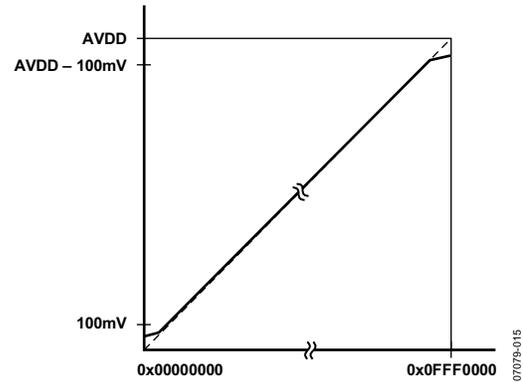


图21. 放大器饱和引起的端点非线性

当有输出负载时, 图21所示的端点非线性会变得更差。在大多数ADuC7060/ADuC7061数据手册中, 正常模式下的DAC输出特性指标都是在DAC输出与一个接地的5 kΩ电阻负载相连的条件下得到的。由于DAC输出被强制提供更多的源电流或吸电流, 图21中的顶部或底部非线性区域将变得更大。而当需要更大电流时, 这会明显地限制输出电压摆幅。

DAC插值模式

与正常模式相比, 在插值模式下, DAC的输出分辨率更高(16位)且更新速率更慢。更新速率通过选择DAC0CON寄存器内的插值时钟速率来控制。在该模式下, 为形成稳定的电压, 需要连接一个外部RC滤波器。

运算放大器模式

在运算放大器模式下, DAC输出缓冲器用作运算放大器, DAC本身禁用。

ADC6是运算放大器的正输入端, ADC7是负输入端, 而ADC8是输出端。在该模式下, 应通过DAC0CON的位9将DAC关断。

非易失FLASH/EE存储器

ADuC7060/ADuC7061片内集成了Flash/EE存储器，可为用户提供非易失、在线可编程存储器空间。

像EEPROM一样，Flash存储器也支持系统内字节编程(尽管必须在向存储器内写入新的数据之前将原有数据擦除)。擦除操作是以页块为单位进行的。因而，通常Flash存储器也被更准确地称作Flash/EE存储器。

总之，由于具有非易失性、在线编程、高密度及低成本等特点，Flash/EE是更理想的存储设备。利用ADuC7060/ADuC7061集成的Flash/EE存储器，用户可以在线更新程序代码空间，而不必在远程操作节点处替换一次性可编程(OTP)设备。

ADuC7060/ADuC7061内置一个32 KB的Flash/EE存储器阵列，其中低30 KB可供用户使用，高2 KB包含永久性嵌入式固件，支持在线串行下载。另外，这2 KB嵌入式固件还包括上电配置程序，用于将出厂标定系数下载到各种已校准的外设，如ADC、温度传感器、带隙基准电压源等。用户代码不允许访问这2 KB嵌入式固件。

FLASH/EE存储器可靠性

器件上的Flash/EE存储器阵列有两个关键的特性：Flash/EE存储器周期耐久性和Flash/EE存储器数据保持力。

耐久性用于衡量Flash/EE存储器重复多个编程、读取及擦除周期的能力。一个单耐久性周期包括4个独立的、连续的事件，定义如下：

- 初始页面擦除时序
- 读/校验时序(单个Flash/EE)
- 存储器的字节编程时序
- 二次读/校验时序(耐久性周期)

在进行可靠性验证时，Flash/EE存储器中的三页(顶、中、底)的每半个字(16位宽)可循环存取10000次(从0x0000至0xFFFF)。Flash/EE存储器耐久力是根据JEDEC保持期限规格A117在-40°C至+125°C工业温度范围内测量出来的。这个结果满足给定温度下最小耐久性超过10,000个周期的规格。

保持期衡量Flash/EE存储器长时间保持编程数据的能力。同样，在具体结温($T_j = 85^\circ\text{C}$)下根据标准JEDEC保持期限规格A117对器件进行测试。作为此测试程序的一部分，如前所述，Flash/EE存储器循环工作的次数到达指定的耐久性限值，然后确定数据保持特性。这就是说，每次对Flash/EE存储器进行重新编程时，Flash/EE存储器在完全指定的保持期限内可以保持数据。注意：基于一个0.6 eV激活能的保持时间随 T_j 递减，如图22所示。

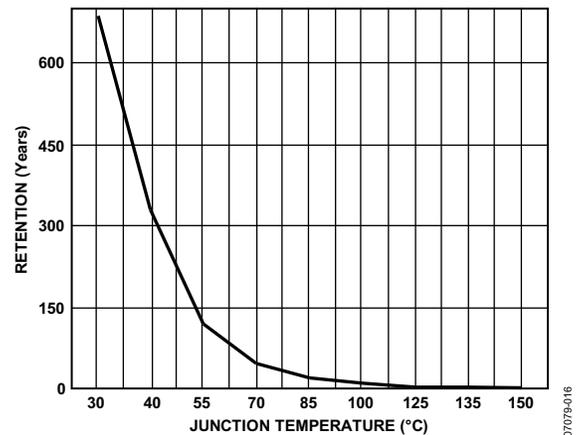


图22. Flash/EE存储器数据保持期

编程

在串行下载模式或JTAG模式下，可对Flash/EE存储器中的30KB区域进行在线编程。

串行下载(在线编程)

ADuC7060/ADuC7061可利用标准UART串行端口下载代码。当器件复位或上电时，如果NTRST/引脚的电平被1 kΩ外接电阻下拉至低电平，则器件将进入串行下载模式。在串行下载模式下，用户可以向Flash/EE的整个30 Kb Flash/EE存储器空间内下载代码(尽管该器件在其目标应用中为在线状态)。由开发系统提供的借助通用异步串口进行串行下载的功能也支持PC串行下载。

当ADuC7060/ADuC7061进入下载模式后，超时周期为2分钟的内部看门狗将被使能。如果Flash擦/写时序无法在该周期内完成，器件将复位。

JTAG访问

JTAG协议允许借助片内JTAG接口进行代码下载和调试。

处理器相关外设

中断系统

ADuC7060/ADuC7061有15个中断源，各个中断源由中断控制器来控制。除用户可编程的软件中断(SWI)外，其余中断均由片内外设产生。ARM7TDMI CPU内核只能识别以下两类中断：正常中断请求(IRQ)和快速中断请求(FIQ)。所有中断都可以被单独屏蔽。

器件内部有许多用于控制和配置中断系统的寄存器。每一个IRQ寄存器和FIQ寄存器中的控制位都代表相同的中断源，如表65所示。

ADuC7060/ADuC7061提供一个矢量中断控制器(VIC)，用于支持中断嵌套(最高可达8级嵌套)。此外，VIC还可以让编程人员为所有中断源指定优先级。通过设置IRQCONN寄存器的ENIRQN位，可以启用中断嵌套功能。当整个矢量中断控制器被使能后，需额外用到大量寄存器。

当内核开始执行中断服务程序(ISR)后，应立即保存IRQSTA/FIQSTA，以确保能够响应所有有效中断。

表65. IRQ/FIQ 寄存器位功能描述

| 位 | 描述 | 备注 |
|----|-----------------------|-------------------------|
| 0 | 所有中断的逻辑或(限FIQ) | 任意FIQ激活时，该位置1。 |
| 1 | 软件中断 | 用户可编程中断源。 |
| 2 | 未定义 | 该位未使用。 |
| 3 | 定时器0 | 通用定时器0。 |
| 4 | 定时器1或唤醒定时器 | 通用定时器1或唤醒定时器。 |
| 5 | 定时器2或看门狗定时器 | 通用定时器2或看门狗定时器。 |
| 6 | 定时器3或STI定时器 | 通用定时器3。 |
| 7 | 未定义 | 该位未使用。 |
| 8 | 未定义 | 该位未使用。 |
| 9 | 未定义 | 该位未使用。 |
| 10 | ADC | ADC中断源位。 |
| 11 | UART | UART中断源位。 |
| 12 | SPI | SPI中断源位。 |
| 13 | XIRQ0 (GPIO IRQ0) | 外部中断0。 |
| 14 | XIRQ1 (GPIO IRQ1) | 外部中断1。 |
| 15 | I ² C主机IRQ | I ² C主机中断源位。 |
| 16 | I ² C从机IRQ | I ² C从机中断源位。 |
| 17 | PWM | PWM触发中断源位。 |
| 18 | XIRQ2 (GPIO IRQ2) | 外部中断2。 |
| 19 | XIRQ3 (GPIO IRQ3) | 外部中断3。 |

IRQ

中断请求(IRQ)是通知处理器进入IRQ模式的一个异常信号。它用于内、外部事件的通用中断处理。

所有的32个位经过逻辑“或”运算后，形成要发送给ARM7TDMI内核的IRQ信号。器件有4个专门用于IRQ的32位寄存器。对各个寄存器的说明如下：

IRQSIG

IRQSIG反映不同IRQ源的状态。如果一个外设产生了一个IRQ信号，IRQSIG中相应的位就会被置1；否则就会被清0。当特定外设的中断请求取消时，IRQSIG的位就会被清0。通过设置IRQEN寄存器，可屏蔽所有IRQ中断源。IRQSIG为只读寄存器。

IRQSIG寄存器

名称： IRQSIG
地址： 0xFFFF0004
默认值： 未定义
访问类型： 只读

IRQEN

IRQEN提供的是当前是能的值，当相应位设为1时，相应的中断源被使能以产生一个IRQ源。IRQEN寄存器不能关断中断，写零没有作用。

IRQEN寄存器

名称： IRQEN
地址： 0xFFFF0008
默认值： 0x00000000
访问类型： 读、写

IRQCLR

IRQCLR是一个只写寄存器，可清除IRQEN寄存器的相应位，从而屏蔽相应的中断源。将该寄存器的某一位置1，会清除IRQEN寄存器的相应位(但不影响其他位)。寄存器IRQEN和IRQCLR配对使用，可以实现独立的使能屏蔽功能，而无需执行原子性读-改-写操作。清零操作对此无影响。

IRQCLR寄存器

| | |
|-------|------------|
| 名称: | IRQCLR |
| 地址: | 0xFFFF000C |
| 默认值: | 0x00000000 |
| 访问类型: | 只写 |

IRQSTA

IRQSTA是一个只读寄存器，提供当前使能的IRQ源的状态(IRQSIG和IRQEN对应位进行逻辑“与”操作)。当置1时，这个源将向ARM7TDMI内核发出一个有效的IRQ中断请求。没有优先级编码器和中断矢量产生。该功能可以在软件中通过一个普通的中断处理程序实现。

IRQSTA寄存器

| | |
|-------|------------|
| 名称: | IRQSTA |
| 地址: | 0xFFFF0000 |
| 默认值: | 0x00000000 |
| 访问类型: | 只读 |

快速中断请求(FIQ)

快速中断请求(FIQ)是进入处理器FIQ模式的一个异常信号。它提供一个短延时的服务于数据传输或通信通道的任务。FIQ接口与IRQ接口相同，此外，它还会提供二级中断(拥有最高优先级)。器件内有4个32位寄存器专门用于FIQ，包括：FIQSIG、FIQEN、FIQCLR和FIQSTA。

FIQSTA的第31至1位通过“或”逻辑运算后，产生FIQ信号。该信号将分别传递给内核以及FIQ和IRQ寄存器的位0(FIQ源)。

逻辑上FIQEN和FIQCLR不允许一个中断源同时被IRQ屏蔽和FIQ屏蔽启用。FIQEN中的某一位被置1会导致IRQEN中的同一位被清0。同样，IRQEN中的某一位被置1也会导致FIQEN中的同一位被清0。一个中断源可以被IRQEN屏蔽和FIQEN屏蔽禁用。

FIQSIG

FIQSIG反映不同FIQ源的状态。如果一个外设产生了一个FIQ信号，FIQSIG中相应的位就会被置1；否则就会被清0。当特定外设的中断请求取消时，FIQSIG的位就会被清0。通过设置FIQEN寄存器，可屏蔽所有FIQ中断源。FIQSIG为只读寄存器。

FIQSIG寄存器

| | |
|-------|------------|
| 名称: | FIQSIG |
| 地址: | 0xFFFF0104 |
| 默认值: | 未定义 |
| 访问类型: | 只读 |

FIQEN

FIQEN提供了当前使能屏蔽值。当某一位被置1时，相应的中断源就会被使能，并产生一个FIQ异常。当某一位被清0时，相应的中断源就会被禁止或屏蔽，此时将无法产生FIQ异常。FIQEN寄存器无法用来禁用中断。

FIQEN寄存器

| | |
|-------|------------|
| 名称: | FIQEN |
| 地址: | 0xFFFF0108 |
| 默认值: | 0x00000000 |
| 访问类型: | 读、写 |

FIQCLR

FIQCLR是一个只写寄存器，可清除FIQEN寄存器的相应位，从而屏蔽相应的中断源。如将该寄存器的某一位置1，清除FIQEN寄存器的相应位(但不影响其他位)。寄存器FIQEN与FIQCLR配合使用，可以实现独立的使能屏蔽功能，而无需执行原子性读-改-写操作。

FIQCLR寄存器

| | |
|-------|------------|
| 名称: | FIQCLR |
| 地址: | 0xFFFF010C |
| 默认值: | 0x00000000 |
| 访问类型: | 只写 |

FIQSTA

FIQSTA是一个只读寄存器，提供当前使能的FIQ源的状态(FIQSIG和FIQEN对应位进行逻辑“与”操作)。当置1时，这个源将向ARM7TDMI内核发出一个有效的FIQ中断请求。没有优先级编码器和中断矢量产生。该功能可以在软件中通过一个普通的中断处理程序实现。

ADuC7060/ADuC7061

FIQSTA寄存器

名称: FIQSTA
地址: 0xFFFF0100
默认值: 0x00000000
访问类型: 只读

可编程中断

由于可编程中断是无法屏蔽的,因此,它们由另外一个寄存器(SWICFG)来控制,通过这个寄存器可以同时写入IRQSTA和IRQSIG寄存器和/或FIQSTA和FIQSIG寄存器。

SWICFG

32位SWICFG寄存器用于设置软件中断,见表66。该寄存器允许控制可编程源中断。

SWICFG寄存器

名称: SWICFG
地址: 0xFFFF0010
默认值: 0x00000000
访问类型: 只写

表66. SWICFG 寄存器位功能描述

| 位 | 描述 |
|------|---|
| 31:3 | 保留。 |
| 2 | 可编程中断FIQ。将该位置1或清0,可将FIQSTA和FIQSIG寄存器的位1置1或清0。 |
| 1 | 可编程中断IRQ。将该位置1或清0,可将IRQSTA和IRQSIG寄存器的位1置1或清0。 |
| 0 | 保留。 |

注意,任何中断信号的有效时间不得少于中断延迟时间,这样才能保证中断信号能够被中断控制器检测到或者被用户在IRQSTA或FIQSTA寄存器中检测到。

矢量中断控制器(VIC)

ADuC7060/ADuC7061 集成一个增强的中断控制系统或矢量中断控制器。通过设置IRQCONN寄存器的位0,可以启用针对IRQ中断源的矢量中断控制器。同样,通过设置IRQCONN寄存器的位1,可以启用针对FIQ中断源的矢量中断控制器。矢量中断控制器在以下几个方面增强了标准IRQ/FIQ中断:

- 矢量中断—允许用户为每个中断源单独定义中断服务程序地址。这可以通过IRQBASE和IRQVEC寄存器来完成。
- IRQ/FIQ中断—可根据优先级进行嵌套,最多允许8级嵌套。FIQ中断的优先级高于IRQ中断。因此,当使能FIQ和IRQ的矢量中断控制器且优先级为最高时,将有可能形成16个不同的中断级。
- 可编程中断优先级—通过设置IRQP0至IRQP2寄存器,可以为中断源分配不同的中断优先级(0至7)。

VIC 寄存器

IRQBASE

IRQBASE(矢量基址寄存器)用于指向存储32位指针地址的存储器的起始地址。这些指针地址是各个终端服务程序的地址。

IRQBASE寄存器

名称: IRQBASE
地址: 0xFFFF0014
默认值: 0x00000000
访问类型: 读、写

表67. IRQBASE 寄存器位功能描述

| 位 | 访问类型 | 初始值 | 描述 |
|-------|------|-----|------------|
| 31:16 | 只读 | 保留 | 读出的数值始终为0。 |
| 15:0 | 读/写 | 0 | 矢量基地址。 |

IRQVEC

IRQVEC(IRQ中断矢量寄存器)指向包含指向一个存储地址,该地址包含当前活动IRQ的中断服务程序的指针。当产生IRQ中断,且已通过设置IRQCONN的位0启用IRQ中断嵌套功能后,该寄存器为只读的。

IRQVEC寄存器

名称: IRQVEC
地址: 0xFFFF001C
默认值: 0x00000000
访问类型: 只读

表68. IRQVEC 寄存器位功能描述

| 位 | 访问类型 | 初始值 | 描述 |
|-------|------|-----|--|
| 31:23 | 只读 | 0 | 读出的数值始终为0。 |
| 22:7 | 只读 | 0 | IRQBASE寄存器值。 |
| 6:2 | 只读 | 0 | 最高优先级IRQ中断源。这些位存储的数据对应0至19其中的一个数字，代表可能的中断源。例如，如果当前活动的优先级最高的IRQ是定时器1，那么这些位的值为[01000]。 |
| 1:0 | 保留 | 0 | 保留位。 |

优先级寄存器

中断优先级寄存器IRQP0、IRQP1和IRQP2为每一个中断源设置各自的中断优先级。中断优先级共分为8级(第0至7级)，其中第0级为最高优先级。

IRQP0寄存器

名称： IRQP0
地址： 0xFFFFF020
默认值： 0x00000000
访问类型： 读、写

表69. IRQP0 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|--------|---------------------|
| 31:27 | 保留 | 保留位。 |
| 26:24 | T3PI | 为定时器3设置中断优先级(0至7)。 |
| 23 | 保留 | 保留位。 |
| 22:20 | T2PI | 为定时器2设置中断优先级(0至7)。 |
| 19 | 保留 | 保留位。 |
| 18:16 | T1PI | 为定时器1设置中断优先级(0至7)。 |
| 15 | 保留 | 保留位。 |
| 14:12 | TOPI | 为定时器0设置中断优先级(0至7)。 |
| 11:7 | 保留 | 保留位。 |
| 6:4 | SWINTP | 为软件中断源设置中断优先级(0至7)。 |
| 3:0 | 保留 | 中断0不能设置优先级。 |

IRQP1寄存器

名称： IRQP1
地址： 0xFFFFF024
默认值： 0x00000000
访问类型： 读、写

表70. IRQP1 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|--------|----------------------------------|
| 31 | 保留 | 保留位。 |
| 30:28 | I2CMPI | 为I ² C主机设置中断优先级(0至7)。 |
| 27 | 保留 | 保留位。 |
| 26:24 | IRQ1PI | 为IRQ1设置中断优先级(0至7)。 |
| 23 | 保留 | 保留位。 |
| 22:20 | IRQ0PI | 为IRQ0设置中断优先级(0至7)。 |
| 19 | 保留 | 保留位。 |
| 18:16 | SPIMPI | 为SPI主机设置中断优先级(0至7)。 |
| 15 | 保留 | 保留位。 |
| 14:12 | UARTPI | 为UART设置中断优先级(0至7)。 |
| 11 | 保留 | 保留位。 |
| 10:8 | ADCPI | 为ADC中断源设置中断优先级(0至7)。 |
| 7:0 | 保留 | 保留位。 |

IRQP2寄存器

名称： IRQP2
地址： 0xFFFFF028
默认值： 0x00000000
访问类型： 读、写

表71. IRQP2 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|--------|----------------------------------|
| 31:15 | 保留 | 保留位。 |
| 14:12 | IRQ3PI | 为IRQ3设置中断优先级(0至7)。 |
| 11 | 保留 | 保留位。 |
| 10:8 | IRQ2PI | 为IRQ2设置中断优先级(0至7)。 |
| 7 | 保留 | 保留位。 |
| 6:4 | SPISPI | 为SPI从机设置中断优先级(0至7)。 |
| 3 | 保留 | 保留位。 |
| 2:0 | I2CSPI | 为I ² C从机设置中断优先级(0至7)。 |

ADuC7060/ADuC7061

IRQCONN

IRQCONN寄存器为IRQ和FIQ控制寄存器，它包含两个有效位：第一位用于使能IRQ中断的嵌套和优先级设置，第二位用于使能FIQ中断的嵌套和优先级设置。

将这两位清0，则FIQ和IRQ始终可用，但无法嵌套IRQ或FIQ。此外，也无法设置中断源的中断优先级。在此默认状态下，FIQ的优先级高于IRQ的优先级。

IRQCONN寄存器

名称： IRQCONN
地址： 0xFFFF003C
默认值： 0x00000000
访问类型： 读、写

表72. IRQCONN 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|--------|---|
| 31:2 | 保留 | 这些位为保留位，不允许向其中写入数据。 |
| 1 | ENFIQN | 将该位置1，可以使能FIQ中断嵌套。将该位清0，则不能设置FIQ中断嵌套及优先级。 |
| 0 | ENIRQN | 将该位置1，可以使能IRQ中断嵌套。将该位清0，则不能设置IRQ中断嵌套及优先级。 |

IRQSTAN

如果IRQCONN[0]被断言且IRQVEC被读取，则这些位中的一位将被断言。IRQ的优先级决定了第几位将被断言。如果IRQ中断的优先级为0，则位0被断言；如果IRQ中断的优先级为1，则位1被断言；以此类推。将该寄存器中的某一位置1，可屏蔽所有与该位所对应的中断具有相同或更低优先级的中断。

如需清0该寄存器中的某一位，必须先将其它与更高优先级对应的位清0。一次只能清0一位。例如，如果该寄存器的值为0x09，第一次写0xFF可以将该寄存器的值改为0x08，再一次写0xFF可以将该寄存器的值改为0x00。

IRQSTAN寄存器

名称： IRQSTAN
地址： 0xFFFF003C
默认值： 0x00000000
访问类型： 读、写

表73. IRQSTAN 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|----|---|
| 31:8 | 保留 | 这些位为保留位，不允许向其中写入数据。 |
| 7:0 | | 将该位置1，可以使能FIQ中断嵌套。将该位清0，则不能设置FIQ中断嵌套及优先级。 |

FIQVEC

FIQVEC(FIQ中断矢量寄存器)指向包含指向一个存储地址，该地址包含当前活动FIQ的中断服务程序的指针。当产生FIQ中断，且已通过设置IRQCONN的位1启用FIQ中断嵌套功能后，该寄存器为只读的。

FIQVEC寄存器

名称： FIQVEC
地址： 0xFFFF011C
默认值： 0x00000000
访问类型： 只读

表74. FIQVEC 寄存器位功能描述

| 位 | 名称 | 初始值 | 描述 |
|-------|----|-----|--|
| 31:23 | 只读 | 0 | 读出的数值始终为0。 |
| 22:7 | 只读 | 0 | IRQBASE寄存器值。 |
| 6:2 | | 0 | 最高优先级FIQ中断源。这些位存储的数据对应0至19其中的一个数字，代表可能的中断源。例如，如果当前活动的优先级最高的FIQ是定时器1，那么这些位的值为[01000]。 |
| 1:0 | 保留 | 0 | 保留位。 |

FIQSTAN

如果IRQCONN[1]被断言且FIQVEC被读取，则这些位中的一位将被断言。FIQ的优先级决定了第几位将被断言。如果FIQ的优先级为0，则位0被断言；如果FIQ中断的优先级为1，则位1被断言；以此类推。

将该寄存器中的某一位置1，可屏蔽所有与该位所对应的中断具有相同或更低优先级的中断。

如需清0该寄存器中的某一位，必须先将其它与更高优先级对应的位清0。一次只能清0一位。例如，如果该寄存器的值为0x09，第一次写0xFF可以将该寄存器的值改为0x08，再一次写0xFF可以将该寄存器的值改为0x00。

FIQSTAN寄存器

名称： FIQSTAN
地址： 0xFFFF013C
默认值： 0x00000000
访问类型： 读、写

表75. FIQSTAN 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|----|---|
| 31:8 | 保留 | 这些位为保留位，不允许向其中写入数据。 |
| 7:0 | | 将该位置1，可以使能FIQ中断嵌套。将该位清0，则不能设置FIQ中断嵌套及优先级。 |

表76. IRQCONE 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|--------------|--|
| 31:8 | 保留 | 这些位为保留位，不允许向其中写入数据。 |
| 7:6 | IRQ3SRC[1:0] | [11] = 外部IRQ3在下降沿触发。 [10] = 外部IRQ3在上升沿触发。 [01] = 外部IRQ3在低电平时触发。 [00] = 外部IRQ3在高电平时触发。 |
| 5:4 | IRQ2SRC[1:0] | [11] = 外部IRQ2在下降沿触发。 [10] = 外部IRQ2在上升沿触发。 [01] = 外部IRQ2在低电平时触发。 [00] = 外部IRQ2在高电平时触发。 |
| 3:2 | IRQ1SRC[1:0] | [11] = 外部IRQ1在下降沿触发。 [10] = 外部IRQ1在上升沿触发。 [01] = 外部IRQ1在低电平时触发。 [00] = 外部IRQ1在高电平时触发。 |
| 1:0 | IRQ0SRC[1:0] | [11] = 外部IRQ0在下降沿触发。 [10] = 外部IRQ0在上升沿触发。 [01] = 外部IRQ0在低电平时触发。 [00] = 外部IRQ0在高电平时触发。 |

外部中断(IRQ0至IRQ3)

ADuC7060/ADuC7061可支持4个外部中断源。这些外部中断源可单独配置为电平触发、上升沿/下降沿触发。

为使能外部中断源，首先必须将FIQEN或IRQEN寄存器的适当位置1。必须正确配置IRQCONE寄存器，才可以选择所需的边沿或电平触发。

设置IRQCLRE中的适当位，可以清除边沿触发的外部IRQ中断。

IRQCONE寄存器

名称： IRQCONE
地址： 0xFFFF0034
默认值： 0x00000000
访问类型： 读、写

ADuC7060/ADuC7061

IRQCLRE寄存器

名称: IRQCLRE
地址: 0xFFFF0038
默认值: 0x00000000
访问类型: 读、写

表77. IRQCLRE 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------|---------------------------------------|
| 31:20 | 保留 | 这些位为保留位, 不允许向其中写入数据。 |
| 19 | IRQ3CLRI | 必须在IRQ3中断服务程序中在该位写入1, 以清除边沿触发的IRQ3中断。 |
| 18 | IRQ2CLRI | 必须在IRQ2中断服务程序中在该位置1, 以清除边沿触发的IRQ2中断。 |
| 17:15 | 保留 | 这些位为保留位, 不允许向其中写入数据。 |
| 14 | IRQ1CLRI | 必须在IRQ1中断服务程序中在该位置1, 以清除边沿触发的IRQ1中断。 |
| 13 | IRQ0CLRI | 必须在IRQ0中断服务程序中在该位置1, 以清除边沿触发的IRQ0中断。 |
| 12:0 | 保留 | 这些位为保留位, 不允许向其中写入数据。 |

定时器

ADuC7060/ADuC7061具有4个通用定时器/计数器。

- 定时器0
- 定时器1或唤醒定时器
- 定时器2或看门狗定时器
- 定时器3

这4个定时器可以在自由模式或周期模式下工作。

当工作在自由模式下时，计数器从最大值或最小值开始递减/递增至0或满量程，然后在从最大值/最小值重新开始递减/递增。

当工作在周期模式下时，计数器以载入寄存器(TxLD 寄存器)中的值为起始值，开始递减/递增计数至0或满量程，然后再以该寄存器中的值为起始值，重新开始计数。注意，在配置TxCON 寄存器前，必须配置TxLD 寄存器。

定时器间隔可通过下式计算：

如果将定时器设置为递减计数，那么

$$\text{间隔} = (\text{TxLD} \times \text{预分频器}) / \text{源时钟}$$

如果将定时器设置为递增计数，那么

$$\text{间隔} = ((\text{满量程} - \text{TxLD}) \times \text{预分频器}) / \text{源时钟}$$

通过访问计数器的值寄存器(TxVAL)，可以随时读出计数器的值。通过对某一定时器的控制寄存器(TxCON)进行写操作，可以启动相应的定时器。

在正常模式下，每当计数器的值达到0(递减计数)或满量程(递增计数)时，都会产生一个IRQ中断。向某一定时器(TxCLR1)的清除寄存器内写入任一数据，可以清除IRQ中断。

表78. 定时器事件捕捉

| 位 | 描述 |
|----|--------------------|
| 0 | 保留 |
| 1 | 定时器0 |
| 2 | 定时器1或唤醒定时器 |
| 3 | 定时器2或看门狗定时器 |
| 4 | 定时器3 |
| 5 | 保留 |
| 6 | 保留 |
| 7 | 保留 |
| 8 | ADC |
| 9 | UART |
| 10 | SPI |
| 11 | XIRQ0 |
| 12 | XIRQ1 |
| 13 | I ² C主机 |
| 14 | I ² C从机 |
| 15 | PWM |
| 16 | XIRQ2 (GPIO IRQ2) |
| 17 | XIRQ3 (GPIO IRQ3) |

时:分:秒:1/128格式

如需使用定时器的时:分:秒:百位数格式，则应选择32768 kHz时钟和256预分频器。百位数字段不代表毫秒，而是一秒的1/128 (256/32768)。代表时、分、秒的位在寄存器中不是连续的。使用TxCON[5:4]设置的“时:分:秒:百位数”格式时，这一配置适用于TxLd和TxVAL。详情参见表79。

表79. 时:分:秒:百位数格式

| 位 | 数值 | 说明 |
|-------|------------|--------|
| 31:24 | 0至23或0至255 | 小时 |
| 23:22 | 0 | 保留 |
| 21:16 | 0至59 | 分钟 |
| 15:14 | 0 | 保留 |
| 13:8 | 0至59 | 秒 |
| 7 | 0 | 保留 |
| 6:0 | 0至127 | 1/128秒 |

定时器0

定时器0是一个带有可编程预分频器的32位通用定时器，可递增计数或递减计数。预分频器源可以是低功耗32.768 kHz振荡器、内核时钟或两个外部通用输入输出端口中的任意一个。预分频系数为1、16、256或32768。这样，在忽略外部通用输入输出端口、预分频系数为1的条件下，定时器的最小分辨率能够达到97.66 ns。

计数器可以采用标准的32位数据格式或者时间格式(小时:分钟:秒:百分之一秒)。

定时器0有一个捕捉寄存器(T0CAP)，它可以被选定的IRQ中断源初始断言所触发。中断触发时，定时器的当前值被复制到T0CAP内，与此同时，定时器继续运行。该功能可用于更精确地判断事件断言。注意：只有IRQ中断源已被使能的外围设备才能使用定时器捕捉功能。

定时器0接口有5个寄存器：T0LD、T0VAL、T0CAP、T0CLRI和T0CON。

- T0LD、T0VAL和T0CAP均为32位寄存器，用于保存32位无符号整数；其中T0VAL和T0CAP为只读寄存器。
- T0CLRI是一个8位寄存器。向该寄存器内写入任意值，可清除定时器0中断。
- T0CON是一个配置寄存器，如表80所示。

定时器0带有一个后分频器，允许用户对定时器0超时次数进行计数，计数范围为1至256。如需要激活后分频器，用户需将T0CON寄存器的位18置1，并向该寄存器的位[24:31]内写入所需的数值。当达到设定的超时次数后，如果T0CON[18]置1，则定时器0就会产生一个中断。

注意，当器件处于低功耗模式且定时器0从GPIO或低功耗振荡器源处接收时钟信号时，定时器0仍继续工作。

当产生定时器0溢出后，定时器0从T0LD寄存器中重新载入数值。

定时器0载入寄存器

名称： T0LD
地址： 0xFFFF0320
默认值： 0x00000000
访问类型： 读、写
功能： T0LD是一个32位寄存器，它保存载入计数器的32位数值。

定时器0清除寄存器

名称： T0CLRI0xFFFF0320
地址： 0xFFFF032C
默认值： 只写
访问类型： 通过用户代码向该8位只写寄存器写入任意值，可以清除中断。

定时器0值寄存器

名称： T0VAL
地址： 0xFFFF0324
默认值： 0xFFFFFFFF
访问类型： 只读
功能： T0VAL是一个32位寄存器，可用于保存定时器0的当前值。

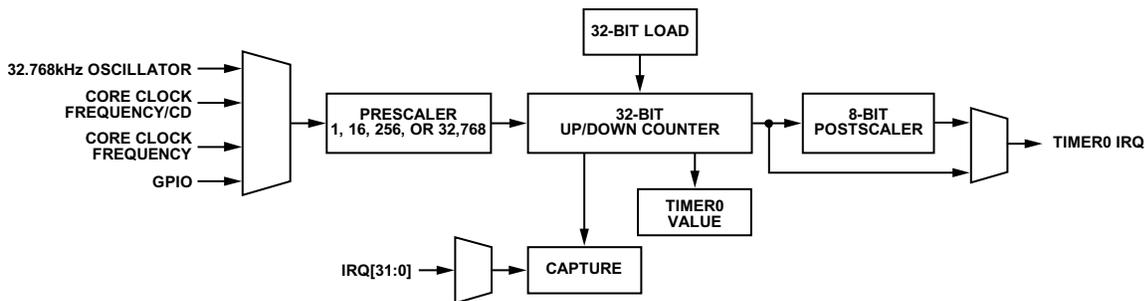


图23. 定时器0框图

07059-017

定时器0捕捉寄存器

名称： T0CAP0xFFFF0518
 地址： 0xFFFF0330
 默认值： 0x00000000
 访问类型： 只读
 功能： 该32位寄存器用于保存被使能IRQ事件所捕捉的32位数值。

定时器0控制寄存器

名称： T0CON
 地址： 0xFFFF0328
 默认值： 0x01000000
 访问类型： 读、写
 功能： 该32位寄存器用于设置定时器0的工作模式。

表80. T0CON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------|--|
| 31:24 | TOPVAL | 8位后分频器。 向该8位写入，可将值写入后分频器。写入0，则视为1。 对这8位进行读操作，得到计数器的当前值。 |
| 23 | TOPEN | 定时器0使能后分频器。 将该位置1，可以使能定时器0后分频器。使能后，在TOLD所定义的T0CON[31:24]个时钟周期后产生中断。 将该位清0，将禁用定时器0后分频器。 |
| 22:20 | | 保留。这些位保留，应由用户代码写入0。 |
| 19 | TOPCF | 后分频器比较标志位，只读。如果定时器0溢出次数等于向后分频器内写入的值，则该位置1。 |
| 18 | T0SRCI | 定时器0中断源。 将该位置1，可选择从后分频器计数器产生中断。 将该位清0，可选择直接从定时器0产生中断。 |
| 17 | T0CAPEN | 事件使能位。 用户置1，可使能对事件的实时捕捉。 用户清0，可禁用对事件的实时捕捉。 |
| 16:12 | T0CAPSEL | 事件选择位[17:0]。事件见表78。 |
| 11 | | 保留位。 |
| 10:9 | T0CLKSEL | 时钟选择。 [00] = 32.768 kHz. [01] = 10.24 MHz/CD. [10] = 10.24 MHz. [11] = P1.0. |
| 8 | T0DIR | 递增计数。 用户置1，定时器0递增计数。 用户清0，定时器0递减计数(默认)。 |
| 7 | T0EN | 定时器0使能位。 用户置1，使能定时器0。 用户清0，禁用定时器0(默认)。 |
| 6 | T0MOD | 定时器0模式。 用户置1，选择周期模式。 用户清0，选择自由模式(默认)。 |

ADuC7060/ADuC7061

| 位 | 名称 | 描述 |
|-----|----------|---|
| 5:4 | T0FORMAT | 数据格式。 [00] = 二进制(默认)。 [01] = 保留。 [10] = 小时: 分钟: 秒: 百分之一秒(23至0小时)。 [11] = 小时: 分钟: 秒: 百分之一秒(255至0小时)。 |
| 3:0 | T0SCALE | 预分频器。 [0000] = 时钟源/1(默认)。 [0100] = 时钟源/16。 [1000] = 时钟源/256。 [1111] = 时钟源/32768。注意: 其它所有值均未定义。 |

定时器1或唤醒定时器

定时器1是一个带有可编程预分频器的32位唤醒定时器，可递增计数或递减计数。预分频器直接从四种时钟源接收时钟信号，即内核时钟(默认选择)、外部32.768 kHz时钟晶振或32.768 kHz振荡器。所选的时钟源可经1、16、256或32768分频。当内核时钟被禁用后，唤醒定时器仍继续工作。这样，当内核在10.24 MHz频率下工作时，如预分频系数为1且忽略外部通用输入输出端口，定时器的最小分辨率能够达到97.66 ns。

计数器可以采用标准的32位数值格式或者时间格式(小时: 分钟: 秒: 百分之一秒)。

当产生定时器1溢出时或当执行T1LD写操作后，定时器1将从T1LD载入数值。

定时器1接口有4个寄存器：

- T1LD和T1VAL均为32位寄存器，用于保存32位无符号整数。T1VAL为只读寄存器。
- T1CLR1是一个8位寄存器。向其中写入任意值，可以清除定时器1中断。
- T1CON是一个配置寄存器，如表81所示。

定时器1载入寄存器

| | |
|-------|--------------------------------|
| 名称: | T1LD |
| 地址: | 0xFFFF0340 |
| 默认值: | 0x00000000 |
| 访问类型: | 读、写 |
| 功能: | T1LD是一个32位寄存器，用于保存载入计数器的32位数值。 |

定时器1清除寄存器

| | |
|-------|------------------------------|
| 名称: | T1CLR1 |
| 地址: | 0xFFFF034C |
| 访问类型: | 只写 |
| 功能: | 通过用户代码向该8位只写寄存器写入任意值，可以清除中断。 |

定时器1值寄存器

| | |
|-------|-------------------------------|
| 名称: | T1VAL |
| 地址: | 0xFFFF0344 |
| 默认值: | 0xFFFFFFFF |
| 访问类型: | 只读 |
| 功能: | T1VAL是一个32位寄存器，可用于保存定时器1的当前值。 |

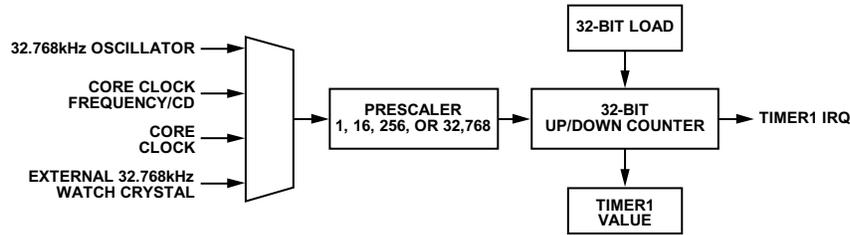


图24. 定时器1框图

定时器1控制寄存器

| | |
|-------|-----------------------|
| 名称: | TICON |
| 地址: | 0xFFFF0348 |
| 默认值: | 0x0000 |
| 访问类型: | 读、写 |
| 功能: | 该16位寄存器用于设置定时器1的工作模式。 |

表81. TICON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------|---|
| 15:11 | | 保留。 |
| 10:9 | T1CLKSEL | 时钟源选择。 [00] = 32.768 kHz振荡器。 [01] = 10.24 MHz/CD。 [10] = XTALI。 [11] = 10.24 MHz。 |
| 8 | T1DIR | 递增计数。 用户置1, 定时器1递增计数。 用户清0, 定时器1递减计数(默认)。 |
| 7 | T1EN | 定时器1使能位。 用户置1, 使能定时器1。 用户清0, 禁用定时器1(默认)。 |
| 6 | T1MOD | 定时器1模式。 用户置1, 选择周期模式。 用户清0, 选择自由模式(默认)。 |
| 5:4 | T1FORMAT | 数据格式。 [00] = 二进制(默认)。 [01] = 保留。 [10] = 小时:分钟:秒:百分之一秒(23至0小时)。只在采用32 kHz时钟时有效。 [11] = 小时:分钟:秒:百分之一秒(255至0小时)。只在采用32 kHz时钟时有效。 |
| 3:0 | T1SCALE | 预分频器。 [0000] = 时钟源/1(默认)。 [0100] = 时钟源/16。 [1000] = 时钟源/256。此设置应采用小时:分钟:秒:百分之一秒格式, 与定时器1配合使用。见表81中所列位[5:4]的格式10和格式11。 [1111] = 时钟源/32768。 |

定时器2或看门狗定时器

定时器2有两种工作模式：正常模式和看门狗模式。看门狗定时器用于强制处理器从非法软件状态恢复到正常工作状态。一旦看门狗定时器被使能，它需要周期性服务来阻止自身强制处理器执行复位操作。

当产生定时器2溢出时或当执行T2CLRI写操作后，定时器2将从T2LD载入数值。

正常模式

在正常模式下的定时器2和在16位工作模式下的定时器0相同(除时钟源不同以外)。时钟源采用低功耗32.768 kHz振荡器，分频系数共有1、16和256三种。

看门狗模式

通过设置T2CON位[5]，进入看门狗模式。定时器2以T2LD寄存器中的超时值作为初始值，进行递减计数，直至计数值为0。当使用1/256预分频时，最大的超时时间为512秒，且为T2LD的满量程。

Flash/EE存储器完成一次页擦除操作和内核执行需要20 ms，为避免与此发生任何冲突，用户软件不得将超时周期设为小于30 ms。

当T2VAL到达0后，根据T2CON[1]的值决定系统产生复位还是中断。为避免产生复位或者中断，必须在T2VAL达到0之前，向T2CLRI内写入任意值。此操作可将T2LD的值重新载入计数器内，且重新开始一个新的超时周期。

一旦进入看门狗模式，T2LD和T2CON就会被写保护。除非通过上电复位将看门狗定时器复位，否则，无法修改这两个寄存器内的数据。其它任何复位事件产生后，看门狗定时器仍会继续计数。为避免无限循环地执行看门狗复位操作，需在用户代码的开始部分对看门狗定时器进行设置。用户软件应将最短超时周期定义为30 ms。

定时器2在JTAG调试访问期间自动暂停，且只有当JTAG放弃对ARM7内核的控制权后才会重新开始计数。默认情况下，定时器2可在断电期间继续计数。为禁用此功能，可将T2CON位0置位。建议采用默认值，即在断电期间仍允许看门狗定时器继续计数。

定时器2接口

定时器2接口有4个寄存器：

- T2CON是一个配置寄存器，如表82所示。
- T2LD和T2VAL均为16位寄存器(位0至15)，用于保存16位无符号整数。T2VAL为只读寄存器。
- T2CLRI是一个8位寄存器。在正常模式下向这个寄存器写入任意值，将清除定时器2中断；在看门狗模式下向这个寄存器写入任意值，则重新开始一个超时周期。

定时器2载入寄存器

名称： T2LD

地址： 0xFFFFF0360

默认值： 0x3BF8

访问类型： 读、写

功能： 该16位寄存器用于保存定时器2的重新载入值。

定时器2清除寄存器

名称： T2CLRI

地址： 0xFFFFF036C

访问类型： 只写

功能： 在看门狗模式下，通过用户代码向该8位只写寄存器写入任意值，可刷新(重新载入)定时器2，以防止产生看门狗定时器复位。

定时器2值寄存器

名称： T2VAL

地址： 0xFFFFF0364

默认值： 0x3BF8

访问类型： 只读

功能： 该16位只读寄存器用于保存定时器2的当前计数值。

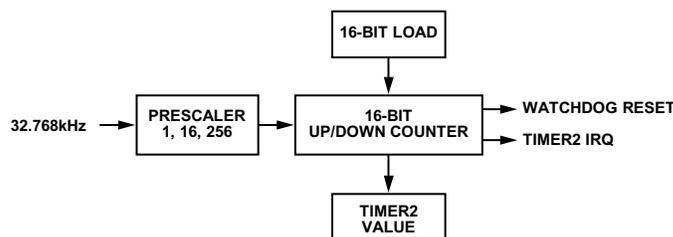


图25. 定时器2框图

定时器2控制寄存器

名称： T2CON

地址： 0xFFFF0368

默认值： 0x0000

访问类型： 读、写

功能： 该16位寄存器用于配置定时器2的工作模式，该寄存器功能的详细说明见表82。

表82. T2CON 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|----------|--|
| 15:9 | | 保留。这些位保留，应由用户代码写入0。 |
| 8 | T2DIR | 递增/递减计数使能。 用户代码置1，可将定时器2的计数方式设定为递增计数。 用户代码清0，可将定时器2的计数方式设定为递减计数。 |
| 7 | T2EN | 定时器2使能。 用户代码置1，使能定时器2。 用户代码清0，禁用定时器2。 |
| 6 | T2MOD | 定时器2工作模式。 用户代码置1，可将定时器2的工作模式设定为周期模式。 用户代码清0，可将定时器2的工作模式设定为自由模式。 |
| 5 | WDOGMDEN | 看门狗定时器模式使能。 用户代码置1，使能看门狗模式。 用户代码清0，禁用看门狗模式。 |
| 4 | | 保留。这些位保留，应由用户代码写入0。 |
| 3:2 | T2SCALE | 定时器2时钟(32.768 kHz)预分频器。 00 = 32.768 kHz (默认)。 01 = 时钟源/16。 10 = 时钟源/256。 11 = 保留。 |
| 1 | WDOGENI | 看门狗定时器IRQ使能。 用户代码置1，在看门狗计数值为0时产生IRQ中断而非复位。 用户代码清0，禁用IRQ选项。 |
| 0 | T2PDOFF | 当断电时，定时器2停止工作。 用户代码置1，在利用POWCON0寄存器的位4使外设断电后，定时器2将停止工作。 用户代码清0，则当利用POWCON0寄存器的位4使外设断电时，使能定时器2。 |

ADuC7060/ADuC7061

定时器3

定时器3是一个带有可编程预分频器的16位通用定时器，可递增计数或递减计数。定时器3采用内核时钟或低功耗32.768 kHz振荡器(预分频系数为1、16、256或32768)作为其时钟源。

定时器3有一个事件捕捉寄存器(T3CAP)，它可以被选定的IRQ中断源初始断言所触发。中断触发时，定时器的当前值将被复制到T3CAP内，与此同时，定时器继续运行。此功能可用于更精确地判断事件断言。

定时器3接口有5个寄存器：

- T3LD、T3VAL和T3CAP均为16位寄存器，用于保存16位无符号整数。T3VAL和T3CAP均为只读寄存器。
- T3CLRI是一个8位寄存器。向其中写入任意值就可以清除中断。
- T3CON是配置寄存器，如表83所示。

定时器3载入寄存器

名称： T3LD
地址： 0xFFFF0380
默认值： 0x0000
访问类型： 读、写
功能： T3LD是一个16位寄存器，它保存载入计数器的16位数值。

定时器3清除寄存器

名称： T3CLRI
地址： 0xFFFF038C
访问类型： 只写
功能： 通过用户代码向该8位只写寄存器写入任意值，可以清除中断。

定时器3值寄存器

名称： T3VAL
地址： 0xFFFF0384
默认值： 0xFFFF
访问类型： 只读
功能： T3VAL是一个16位寄存器，可用于保存定时器3的当前值。

定时器3捕捉寄存器

名称： T3CAP
地址： 0xFFFF0390
默认值： 0x0000
访问类型： 只读
功能： 这是一个16位寄存器，用于保存使能IRQ事件所捕捉的16位数值。

定时器3控制寄存器

名称： T3CON
地址： 0xFFFF0388
默认值： 0x00000000
访问类型： 读、写
功能： 该32位寄存器用于设置定时器3的工作模式。

表83. T3CON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------|--|
| 31:18 | | 保留。 |
| 17 | T3CAPEN | 事件使能位。 用户置1, 可使能对事件的实时捕捉。 用户清0, 可禁用对事件的实时捕捉。 |
| 16:12 | T3CAPSEL | 事件选择范围(0至17)。事件如表78所示。 |
| 11 | | 保留。 |
| 10:9 | T3CLKSEL | 时钟选择。 [00] = 32.768 kHz振荡器。 [01] = 10.24 MHz/CD。 [10] = 10.24 MHz。 [11] = 保留。 |
| 8 | T3DIR | 递增计数。 用户置1, 可将定时器3的计数方式设定为递增计数。 用户清0, 可将定时器3的计数方式设定为递减计数(默认)。 |
| 7 | T3EN | 定时器3使能位。 用户置1, 可使能定时器3。 用户清0, 可禁用定时器3(默认)。 |
| 6 | T3MOD | 定时器3工作模式。 用户置1, 选择周期模式。 用户清0, 选择自由模式(默认模式)。 |
| 5:4 | | 保留。 |
| 3:0 | T3SCALE | 预分频器。 [0000] = 时钟源/1(默认)。 [0100] = 时钟源/16。 [1000] = 时钟源/256。 [1111] = 时钟源/32768。 |

脉宽调制器

脉宽调制器概述

ADuC7060/ADuC7061 集成了一个6通道脉宽调制器(PWM)接口。PWM输出即可用于驱动H桥或也可作为标准PWM输出。上电后，PWM输出的默认为H桥。这可以确保在默认状态下电机是关闭的。在标准PWM模式下，从三对PWM引脚内输出信号。用户既可以控制每一对输出引脚的工作周期，又可以单独控制每一个输出端的占空比。

表84. PWM 寄存器

| 寄存器名称 | 描述 |
|----------|---------------------------|
| PWMCON | PWM控制。 |
| PWM0COM0 | 用于比较PWM输出0和PWM输出1的比较寄存器0。 |
| PWM0COM1 | 用于比较PWM输出0和PWM输出1的比较寄存器1。 |
| PWM0COM2 | 用于比较PWM输出0和PWM输出1的比较寄存器2。 |
| PWM0LEN | 用于PWM输出0和PWM输出1的频率控制。 |
| PWM1COM0 | 用于比较PWM输出2和PWM输出3的比较寄存器0。 |
| PWM1COM1 | 用于比较PWM输出2和PWM输出3的比较寄存器1。 |
| PWM1COM2 | 用于比较PWM输出2和PWM输出3的比较寄存器2。 |
| PWM1LEN | 用于PWM输出2和PWM输出3的频率控制。 |
| PWM2COM0 | 用于比较PWM输出4和PWM输出5的比较寄存器0。 |
| PWM2COM1 | 用于比较PWM输出4和PWM输出5的比较寄存器1。 |
| PWM2COM2 | 用于比较PWM输出4和PWM输出5的比较寄存器2。 |
| PWM2LEN | 用于PWM输出4和PWM输出5的频率控制。 |
| PWMCLR1 | PWM中断清除寄存器。 |

PWMxCOMx 寄存器用于控制在各种模式下改变PWM输出状态的时间点。第一对PWM输出示例(PWM0和PWM1)如图26所示。

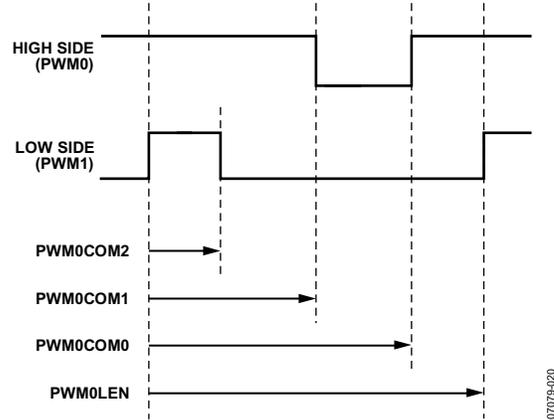


图26. PWM时序

通过PWMCON，可将PWM时钟频率设定为以下值之一：UCLK除以2/4/8/16/32/64/128/256。PWMxLEN用于设定PWM周期长度。

PWM波形由16位定时器的计数值和比较寄存器的内容来决定，如图26所示的PWM0和PWM1的波形。

当定时器计数值达到PWM0LEN时，低端波形(PWM1)变为高电平；当定时器计数值达到PWM0COM2内所保存的数值或者当高端波形(PWM0)变为低电平时，PWM1变为低电平。

当定时器计数值达到PWM0COM0内所保存的数值时，高端波形(PWM0)变为高电平；当定时器计数值达到PWM0COM1内所保存的数值时，PWM0变为低电平。

PWMCON控制寄存器

名称： PWMCON

地址： 0xFFFF0F80

默认值： 0x0012

访问类型： 读、写

功能： 该16位寄存器用于设置PWM输出。

表85. PWMCON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|------------|--|
| 15 | 保留 | 该位为保留位。不可对该位进行写操作。 |
| 14 | Sync | 使能PWM同步功能。 用户置1, 那么, 当检测到P1.2/SYNC引脚上的由高到低变换后, 所有PWM计数器将在下一个时钟沿复位。 用户置0, 则可以忽略在P1.2/SYNC引脚上发生的高低变换。 |
| 13 | PWM5INV | 用户置1, 可以将PWM5反相。 用户清0, PWM5在正常模式下工作。 |
| 12 | PWM3INV | 用户置1, 可以将PWM3反相。 用户清0, PWM3在正常模式下工作。 |
| 11 | PWM1INV | 用户置1, 可以将PWM1反相。 用户清0, PWM1在正常模式下工作。 |
| 10 | PWMTRIP | 用户置1, 可使能PWM触发中断。当PWM触发中断(P1.3/TRIP引脚)处于低电平时, PWMEN位被清0, 且产生中断。 用户清0, 可禁用PWMTRIP中断。 |
| 9 | ENA | 当HOFF = 0且HMODE = 1时, 该位可用。注意: 该位仅在H桥模式下有效。 用户置1, 可使能PWM输出。 用户清0, 可禁用PWM输出。 如HOFF = 1且HMODE = 1, 参见表86。 |
| 8:6 | PWMCP[2:0] | PWM时钟预分频器位。设置UCLK分频数。 [000] = UCLK/2。 [001] = UCLK/4。 [010] = UCLK/8。 [011] = UCLK/16。 [100] = UCLK/32。 [101] = UCLK/64。 [110] = UCLK/128。 [111] = UCLK/256。 |
| 5 | POINV | 用户置1, 可反相所有PWM输出。 用户清0, 则允许正常的PWM输出。 |
| 4 | HOFF | 屏蔽高端。 用户置1, 可强制PWM0和PWM2输出高电平信号。这样做的结果是强制PWM1和PWM3输出低电平信号。 用户清0, 则允许正常的PWM输出。 |
| 3 | LCOMP | 载入比较寄存器。 用户置1后, 在PWM定时器进行从0x00到0x01的下次变换时, PWMxCOMx的值将被载入内部比较寄存器。 用户清0, 以允许使用保存在内部比较寄存器中的数值。 |
| 2 | DIR | 方向控制。 用户置1, 则当PWM2和PWM3处于低电平状态时, 使能PWM0和PWM1作为输出信号。用户置0, 则当PWM0和PWM1处于低电平状态时, 使能PWM2和PWM3作为输出信号。 |
| 1 | HMODE | 启用H桥模式 ¹ 。 用户置1, 以启用H桥模式、使能PWMCON寄存器的位1至位5。 用户清0, 以便让PWM在标准模式下工作。 |
| 0 | PWMEN | 用户置1, 可使能全部PWM输出。 用户清0, 可禁用全部PWM输出。 |

¹ 在H桥模式下, HMODE = 1。关于PWM输出选择的说明, 见表86。

ADuC7060/ADuC7061

上电后，PWMCON寄存器的默认值为0x0012(HOFF = 1、HMODE = 1)。在默认状态下，PWM相关的所有GPIO引脚均被设定为在PWM模式下工作(见表86)。通过向PWMCLRI

寄存器写入任意值，可以清除PWM触发中断。注意：当使用PWM触发中断时，在退出中断服务程序前，将清除PWM中断。这样可以防止同时产生多个中断。

表86. PWM输出选择

| PWMCON 寄存器 ¹ | | | | PWM输出 ² | | | |
|-------------------------|------|-------|-----|--------------------|------|------|------|
| ENA | HOFF | POINV | DIR | PWM0 | PWM1 | PWM2 | PWM3 |
| 0 | 0 | X | X | 1 | 1 | 1 | 1 |
| X | 1 | X | X | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | HS1 | LS1 |
| 1 | 0 | 0 | 1 | HS1 | LS1 | 0 | 0 |
| 1 | 0 | 1 | 0 | HS1 | LS1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | HS1 | LS1 |

¹ X表示无关。

² HS = 高端；LS = 低端。

表87. 比较寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|----------|------------|--------|------|
| PWM0COM0 | 0xFFFF0F84 | 0x0000 | 读/写 |
| PWM0COM1 | 0xFFFF0F88 | 0x0000 | 读/写 |
| PWM0COM2 | 0xFFFF0F8C | 0x0000 | 读/写 |
| PWM1COM0 | 0xFFFF0F94 | 0x0000 | 读/写 |
| PWM1COM1 | 0xFFFF0F98 | 0x0000 | 读/写 |
| PWM1COM2 | 0xFFFF0F9C | 0x0000 | 读/写 |
| PWM2COM0 | 0xFFFF0FA4 | 0x0000 | 读/写 |
| PWM2COM1 | 0xFFFF0FA8 | 0x0000 | 读/写 |
| PWM2COM2 | 0xFFFF0FAC | 0x0000 | 读/写 |

PWM0COM0比较寄存器

名称: PWM0COM0
 地址: 0xFFFF0F84
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM0输出引脚变为高电平。

PWM0COM1比较寄存器

名称: PWM0COM1
 地址: 0xFFFF0F88
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM0输出引脚变为低电平。

PWM0COM2比较寄存器

名称: PWM0COM2
 地址: 0xFFFF0F8C
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM1输出引脚变为低电平。

PWM0LEN寄存器

名称: PWM0LEN
 地址: 0xFFFF0F90
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM1输出引脚变为高电平。

PWM1COM0比较寄存器

名称: PWM1COM0
 地址: 0xFFFF0F94
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM2输出引脚变为高电平。

PWM1COM1比较寄存器

名称: PWM1COM1
 地址: 0xFFFF0F98
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM2输出引脚变为低电平。

PWM1COM2比较寄存器

名称: PWM1COM2
 地址: 0xFFFF0F9C
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM3输出引脚变为低电平。

PWM1LEN寄存器

名称: PWM1LEN
 地址: 0xFFFF0FA0
 默认值: 0x0000
 访问类型: 读、写
 功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM3输出引脚变为高电平。

ADuC7060/ADuC7061

PWM2COM0比较寄存器

名称: PWM2COM0
地址: 0xFFFF0FA4
默认值: 0x0000
访问类型: 读、写
功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM4输出引脚变为高电平。

PWM2COM1比较寄存器

名称: PWM2COM1
地址: 0xFFFF0FA8
默认值: 0x0000
访问类型: 读、写
功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM5输出引脚变为低电平。

PWM2COM2比较寄存器

名称: PWM2COM2
地址: 0xFFFF0FAC
默认值: 0x0000
访问类型: 读、写
功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM5输出引脚变为低电平。

PWM2LEN寄存器

名称: PWM2LEN
地址: 0xFFFF0FB0
默认值: 0x0000
访问类型: 读、写
功能: 当PWM定时器计数达到该寄存器所保存的计数值时, PWM5输出引脚变为高电平。

PWMCLRI寄存器

名称: PWMCLRI
地址: 0xFFFF0FB8
默认值: 0x0000
访问类型: 只写
功能: 向其中写入任意值就可以清除PWM中断源。在退出PWM中断服务程序前, 必须对该寄存器进行写操作; 否则, 将同时产生多个中断。

UART串行接口

ADuC7060/ADuC7061 集成一个16450兼容的UART。该UART是一个全双工通用异步接收器/发送器。UART的作用是对从外设接收的数据字符进行串并转换、对从ARM7TDMI处接收的数据字符进行并串转换。该UART有一个小数分频器，能够促进高精度波特率的生成且支持网络可寻址模式。ADuC7060/ADuC7061的P1.0/IRQ1/SIN/T0和P1.1/SOUT引脚支持UART功能。

串行通信支持异步协议，支持各种字长、停止位以及奇偶校验生成等选项(在配置寄存器内进行选择)。

波特率生成

ADuC7060/ADuC7061提供两种UART波特率生成模式：正常450 UART波特率生成模式和ADuC7060/ADuC7061小数分频器波特率生成模式。

正常450 UART波特率生成

正常456 UART波特率是内核时钟的一个分频，采用COMDIV0和COMDIV1寄存器中的值(16位值，分频锁存器(DL))。标准波特率生成公式为：

$$\text{波特率} = \frac{10.24 \text{ MHz}}{16 \times 2 \times DL} \quad (1)$$

表88列举了几种常见波特率值。

表88.利用标准波特率发生器的波特率

| 波特率 | DL | 实际波特率 | 百分比误差 |
|---------|------|---------|-------|
| 9600 | 0x21 | 9696 | 1.01% |
| 19,200 | 0x11 | 18,824 | 1.96% |
| 115,200 | 0x3 | 106,667 | 7.41% |

ADuC7060/ADuC7061小数分频器

将小数分频器与正常波特率发生器配合使用，可以生成精确、高速的波特率。

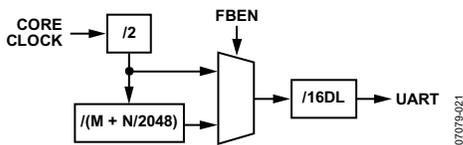


图27. 小数分频器波特率生成

利用小数分频器计算波特率的公式如下：

$$\text{波特率} = \frac{10.24 \text{ MHz}}{16 \times DL \times 2 \times \left(M + \frac{N}{2048}\right)} \quad (2)$$

$$M + \frac{N}{2048} = \frac{10.24 \text{ MHz}}{\text{Baud Rate} \times 16 \times DL \times 2}$$

表89列举了几种常见波特率值。

表89.利用小数波特率发生器的波特率

| 波特率 | DL | M | N | 实际波特率 | 百分比误差 |
|---------|------|---|-----|---------|--------|
| 9600 | 0x21 | 1 | 21 | 9598.55 | 0.015% |
| 19,200 | 0x10 | 1 | 85 | 19,203 | 0.015% |
| 115,200 | 0x2 | 1 | 796 | 115,218 | 0.015% |

UART寄存器定义

UART接口包括下列11个寄存器：

COMTX：8位发送寄存器。

COMRX：8位接收寄存器。

COMDIV0：分频锁存器(低字节)。

COMDIV1：分频锁存器(高字节)。

COMCON0：线路控制寄存器。

COMCON1：线路控制寄存器。

COMSTA0：线路状态寄存器。

COMSTA1：线路状态寄存器。

COMIEN0：中断使能寄存器。

COMIID0：中断识别寄存器。

COMDIV2：16位小数波特率分频寄存器。

COMTX、COMRX和COMDIV0共用相同的地址。当寄存器COMCON0的位7清0时，允许访问寄存器COMTX和COMRX。当寄存器COMCON0或COMCON1的位7置1时，允许访问COMDIV0或COMDIV1。

ADuC7060/ADuC7061

UART发送寄存器。

向该8位寄存器(COMTX)写入数据后，允许利用UART发送数据。

COMTX寄存器

名称： COMTX
地址： 0xFFFF0700
访问类型： 只写

UART接收寄存器

对该8位寄存器进行读操作，以接收UART发送的数据。

COMRX寄存器

名称： COMRX
地址： 0xFFFF0700
默认值： 0x00
访问类型： 只读

UART分频锁存器寄存器0

该8位寄存器(COMDIV0)保持控制UART波特率的分频锁存器的最低有效字节。

COMDIV0寄存器

名称： COMDIV0
地址： 0xFFFF0700
默认值： 0x00
访问类型： 读、写

UART分频锁存器寄存器1

该8位寄存器保持控制UART波特率的分频锁存器的最高有效字节。

COMDIV1寄存器

名称： COMDIV1
地址： 0xFFFF0704
默认值： 0x00
访问类型： 读、写

UART控制寄存器0

该8位寄存器(COMCON0)与COMCON1配合使用，可控制UART操作。

COMCON0寄存器

名称： COMCON0
地址： 0xFFFF070C
默认值： 0x00
访问类型： 读、写

表90. COMCON0 寄存器位功能描述

| 位 | 位 | 描述 |
|-----|------|--|
| 7 | DLAB | 分频锁存器访问。 用户置1，可允许访问寄存器COMDIV0和COMDIV1。 用户清0，禁止访问寄存器COMDIV0和COMDIV1但允许访问COMRX、COMTX和COMIEN0。 |
| 6 | BRK | 设置断开。 用户置1，可强制发送为0。 用户清0，采用正常模式工作。 |
| 5 | SP | 强制奇偶校验。用户置1，强制奇偶校验为定义值。 如果EPS = 1且PEN = 1，则置1。 如果EPS = 0且PEN = 1，则清0。 |
| 4 | EPS | 偶校验选择位。 置1，以选择偶校验。 清0，以选择奇校验。 |
| 3 | PEN | 奇偶校验使能位。 用户置1，以发送并检查奇偶校验位。 用户清0，则既不发送也不检查奇偶校验。 |
| 2 | Stop | 停止位。 用户置1后，如果字长为5位则传送1.5个停止位；如果字长为6、7或8位，则传送2个停止位。 不论所选停止位的个数是多少，接收器只检查第一个停止位。 用户清0，则在发送的数据中只产生一个停止位。 |
| 1:0 | WLS | 字长选择。 [00] = 5位。 [01] = 6位。 [10] = 7位。 [11] = 8位。 |

ADuC7060/ADuC7061

UART控制寄存器1

该8位寄存器与COMCON0配合使用，可控制UART操作。

COMCON1寄存器

位: COMCON1

地址: 0xFFFF0710

默认值: 0x00

访问类型: 读、写

表91. COMCON1 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|----------|--|
| 7:5 | | 保留位。未使用。 |
| 4 | LOOPBACK | 回送。用户置1，使能回送模式。在回送模式下，发送引脚被强制处于高电平的状态。 |
| 3:2 | | 保留位。未使用。 |
| 1 | RTS | 发送请求位。 用户置1，可强制RTS输出为0。 用户清0，可强制RTS输出为1。 |
| 0 | DTR | 数据终端就绪位。 用户置1，可强制DTR输出为0。 用户清0，可强制DTR输出为1。 |

UART状态寄存器0

COMSTA0寄存器

位: COMSTA0

地址: 0xFFFF0714

默认值: 0x60

访问类型: 只读

功能: 该8位只读寄存器用于反映UART的当前状态。

表92. COMSTA0 寄存器位功能描述

| 位 | 名称 | 描述 |
|---|------|---|
| 7 | | 保留。 |
| 6 | TEMT | COMTX和移位寄存器空状态位。 当COMTX和移位寄存器为空时，该位自动置1。该位指示数据已发送；即移位寄存器不再保持数据。 如果COMTX被写入数据，该位自动清0。 |
| 5 | THRE | COMTX空状态位。 当COMTX为空时，该位自动置1。该位置1便可向COMTX内写入数据；但前面数据可能还未发送，且可能仍保存在移位寄存器内。 如果COMTX被写入数据，该位自动清0。 |
| 4 | BI | 断开指示符。 当P1.0/IRQ1/SIN/T0引脚处于低电平的时间超过最大字长时，该位被置1。 该位自动清0。 |
| 3 | FE | 帧错误。 当停止位无效时，该位被置1。 该位自动清0。 |
| 2 | PE | 奇偶校验错误。 当产生奇偶校验错误时，该位被置1。 该位自动清0。 |
| 1 | OE | 溢出错误。 如当前数据在读取前被覆盖，该位自动置1。 该位自动清0。 |
| 0 | DR | 数据就绪。 当COMRX满时，该位自动置1。 COMRX读取后，该位清0。 |

UART状态寄存器1**COMSTA1寄存器**

位: COMSTA1

地址: 0xFFFF0718

默认值: 0x00

访问类型: 只读

功能: COMSTA1为调制解调器状态寄存器。

表93. COMSTA1 寄存器位功能描述

| 位 | 位 | 描述 |
|-----|------|--|
| 7:5 | | 保留。未使用。 |
| 4 | CTS | 清除发送。 |
| 3:1 | | 保留。未使用。 |
| 0 | DCTS | ΔCTS。 在上一次读取COMSTA1后，如果CTS改变状态，则DCTS自动置1。 读取COMSTA1后，该位自动清0。 |

UART中断使能寄存器0**COMIEN0寄存器**

位: COMIEN0

地址: 0xFFFF0704

默认值: 0x00

访问类型: 读、写

功能: 该8位寄存器用于使能或禁用单独的UART中断源。

表94. COMIEN0 寄存器位功能描述

| 位 | 位 | 描述 |
|-----|-------|---|
| 7:4 | | 保留。未使用。 |
| 3 | EDSSI | 调制解调器状态中断使能位。 用户置1后，当COMSTA1[4]或COMSTA1[0]被置1时，允许产生中断。 由用户清0。 |
| 2 | ELSI | 接收状态中断使能位。 用户置1后，当COMSTA0[3:1]寄存器任一位被置1时，使能中断产生。 由用户清0。 |
| 1 | ETBEI | 使能发送缓冲空中断。 用户置1后，在发送期间，如果缓冲器为空(即COMSTA0[5]置1)，则使能中断。 由用户清0。 |
| 0 | ERBFI | 使能接收缓冲满中断。 用户置1后，在接收期间，当缓冲器满时，使能中断。 由用户清0。 |

UART中断识别寄存器0**COMIID0寄存器**

位: COMIID0

地址: 0xFFFF0708

默认值: 0x01

访问类型: 只读

功能: 该8位寄存器用于反映UART中断源。

ADuC7060/ADuC7061

表95. COMIID0 寄存器位功能描述

| 状态位 [2:1] | 位0 | 优先级 | 定义 | 清除操作 |
|-----------|----|-----|-----------|----------------------|
| 00 | 1 | | 无中断 | |
| 11 | 0 | 1 | 接收线路状态中断 | 读取COMSTA0 |
| 10 | 0 | 2 | 接收缓冲满中断 | 读取COMRX |
| 01 | 0 | 3 | 发送缓冲空中断 | 将数据写入COMTX或读取COMIID0 |
| 00 | 0 | 4 | 调制解调器状态中断 | 读COMSTA1寄存器 |

表96. COMDIV2 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|-----------|---|
| 15 | FBEN | 小数波特率发生器使能位。用户置1，使能小数波特率发生器。用户清0，以利用标准450 UART波特率发生器来生成波特率。 |
| 14:13 | | 保留。 |
| 12:11 | FBM[1:0] | M值。当FBM = 0时，M = 4。利用小数分频器计算波特率参见公式2；常见波特率值见表88。 |
| 10:0 | FBN[10:0] | N值。利用小数分频器计算波特率参见公式2；常见波特率值见表88。 |

UART小数分频器寄存器

该16位寄存器COMDIV2用于控制ADuC7060/ADuC7061小数分频器操作。

COMDIV2寄存器

名称： COMDIV2
 地址： 0xFFFF072C
 默认值： 0x0000
 访问类型： 读、写

I²C

ADuC7060/ADuC7061集成了一个I²C外设，用户可将该设备配置成完全I²C兼容的I²C总线主机或者完全I²C总线兼容的从机。引脚SDA和SCL用于数据传输，通过对这两个引脚进行“线与”配置，可以在多主机系统中进行仲裁。这两个引脚需要接外部上拉电阻。上拉电阻的典型电阻值的范围为4.7 kΩ至10 kΩ。

用户可对I²C总线系统内的I²C总线外设进行编程。当发送器不工作时，可随时修改这个ID。用户可以对接口进行配置，以使其响应四个从机地址。

I²C系统的传输过程为：主机在总线处于空闲状态时通过产生起始条件来初始化传输；在初始地址发送期间，主机发送从机的地址和数据传输方向(read or write)；如果主机没有仲裁失效且从机进行了应答，那么开始向从机传输数据；传输会持续到主机发送一个停止条件为止，然后总线进入空闲状态。

在同一时刻，I²C外设无法既作为主机又作为从机。同一个I²C通道不能同时支持主模式和从模式。

ADuC7060/ADuC7061的I²C接口具备以下特性：

- 支持重复起始条件。在主机模式下，通过对ADuC7060/ADuC7061进行编程，可以允许ADuC7060/ADuC7061重复启动一次。在从机模式下，ADuC7060/ADuC7061可识别重复起始条件。
- 在主机和从机模式下，器件可识别7位和10位总线地址。

```
GP0CON0 = BIT4 + BIT12;    // Select SPI/I2C alternative function for P0.1 and P0.3
GP0KEY1 = 0x7;            // Write to GP0KEY1
GP0CON1 = BIT1;          // Select I2C functionality for P0.1 and P0.3
GP0KEY2 = 0x13;          // Write to GP0KEY2
```

- 在I²C主机模式下，ADuC7060/ADuC7061允许在一个传输时序下，从单个从机中连续读取512字节数据。
- 总线上的其它器件可以使能时钟延展，这不会给ADuC7060/ADuC7061带来任何问题。不过，ADuC7060/ADuC7061无法使能时钟延展。
- 在从机模式下，通过对ADuC7060/ADuC7061进行编程，可以允许其返回一个非应答信号(NACK)。这样，可以保证在I²C数据传输结束时校验和字节是有效的。
- 在主机模式下，支持总线仲裁。
- 支持内、外部回送，用于I²C硬件测试。
- 在主、从机模式下，发送和接收电路均有两字节的FIFO缓存。为用户提供状态位，以便控制上述FIFO缓存。

I²C功能的外部引脚配置

ADuC7060/ADuC7061的P0.1/SCLK/SCL引脚和P0.3/MOSI/SDA引脚中支持I²C功能的是P0.1和P0.3。P0.1的功能是传输I²C时钟信号(SCL)，而P0.3的功能是传输I²C数据信号(SDA)。为将P0.1和P0.3用于I²C模式，需将GP0CON0寄存器的位4和位12分别置1。另外，GP0CON1寄存器的位1必须为1，以使能I²C模式。

注意：在对GP0CON1寄存器进行写操作前，必须将GP0KEY1寄存器的值设置为0x7。在对GP0CON1寄存器进行写操作之后，必须立刻将GP0KEY2寄存器的值设置为0x13。详见下列代码示例：

ADuC7060/ADuC7061

串行时钟生成

系统中的I²C主机生成发送串行时钟。主机通道经过配置，可以在快速模式(400 kHz)或标准模式(100 kHz)下工作。

I2CDIV 寄存器中的比特率的定义如下：

$$f_{SERIAL\ CLOCK} = \frac{f_{UCLK}}{(2 + DIVH) + (2 + DIVL)}$$

其中：

f_{UCLK} 是分频之前的时钟。

DIVH是时钟高周期。

DIVL是时钟低周期。

因而，如果希望串行时钟为100 kHz，那么

应该配置 $DIVH = DIVL = 0x33$

如果希望串行时钟为400 kHz，

应该配置 $DIVH = 0x0A, DIVL = 0x0F$

I2CDIV寄存器与DIVH:DIVL相对应。

I²C总线地址

从机模式

在从机模式下，I2CID0、I2CID1、I2CID2和I2CID3寄存器包含器件ID。器件将4个I2CIDx寄存器的内容与从总线主机处接收的地址字节相比较。为确保寻址准确，每一个ID寄存器的7个MSB必须与最先接收到的地址字节的7个MSB相同。在地址识别过程中，ID寄存器的LSB(发送器的方向位)被忽略。

ADuC7060/ADuC7061还支持10位寻址模式。当I2CSCON寄存器的位1(ADR10EN)置1时，在从机模式下，系统支持10位地址，且将该地址保存在I2CID0寄存器和I2CID1寄存器之中。10位地址的组成如下：

I2CID0[0]：读/写位，不属于I²C地址。

I2CID0[7:1] = 地址位[6:0]。

I2CID1[2:0] = 地址位[9:7]。

I2CID1[7:3]的值必须为11110b。

主机模式

在主机模式下，I2CADR0寄存器用于存储器件的I²C地址。

在7位地址模式下，I2CADR0[7:1]用于存储器件地址。I2CADR0[0]是读/写位。

在10位地址模式下，10位地址的组成如下：

I2CADR0[7:3]的值必须为11110b。

I2CADR0[2:1] = 地址位[9:8]。

I2CADR1[7:0] = 地址位[7:0]。

I2CADR0[0]是读/写位。

I²C寄存器

I²C外设接口由19个寄存器组成。其中，9个寄存器只与主机相关，9个寄存器只与从机相关，剩下的一个寄存器为主、从机共用。

I²C主机寄存器

I2CMCON(I²C主机控制寄存器)

名称： I2CMCON

地址： 0xFFFF0900

默认值： 0x0000

访问类型： 读、写

功能： 该16位寄存器用于在主机模式下配置I²C外设。

表97. I2CMCON 寄存器位功能描述

| 位 | 位 | 描述 |
|------|------------|--|
| 15:9 | | 保留。这些位为保留位，不允许向其中写入数据。 |
| 8 | I2CMCENI | I ² C发送完成中断使能位。 将该位置1，则当从I ² C总线上检测到停止条件时，产生中断。 清除该中断源。 |
| 7 | I2CNACKENI | I ² C非应答信号(NACK)接收中断使能位。 置1，则当I ² C主机接收到非应答信号时，产生中断。 清除该中断源。 |
| 6 | I2CALENI | I ² C仲裁失效中断使能位。 置1，则当I ² C主机没有获得对I ² C总线的控制权时，产生中断。 清除该中断源。 |
| 5 | I2CMTENI | I ² C发送中断使能位。 置1，则当I ² C主机完成一个字节的传输后，产生中断。 清除该中断源。 |
| 4 | I2CMRENI | I ² C接收中断使能位。 置1，则当I ² C主机接收到数据时，产生中断。 用户清0，则在I ² C主机接收数据的过程中，禁用中断。 |
| 3 | | 保留。应在此位写入0值。 |
| 2 | I2CILEN | I ² C内部回送使能位。 该位置1，可以使能回送测试模式。在该模式下，SCL和SDA信号在内部分别与各自的输入信号相连。 用户清0，可禁用回送模式。 |
| 1 | I2CBD | I ² C主机撤回禁用位。 置1，可允许器件与另一器件争夺对总线的控制权(即使另一个器件正在产生一个起始条件)。 清0，则在I ² C总线释放，执行撤回。 |
| 0 | I2CMEN | I ² C主机使能位。 用户置1，使能I ² C主机模式。 用户清0，禁用I ² C主机模式。 |

ADuC7060/ADuC7061

I2CMSTA(I²C主机状态寄存器)

名称: I2CMSTA

地址: 0xFFFF0904

默认值: 0x0000

访问类型: 只读

功能: 该16位寄存器是主机模式下的I²C状态寄存器。

表98. I2CMSTA 寄存器位功能描述

| 位 | 位 | 描述 |
|-------|-----------|--|
| 15:11 | | 保留。这些位是保留的。 |
| 10 | I2CBBUSY | I ² C总线忙碌状态位。 当从I ² C总线上检测到起始条件后, 该位被置1。 当从I ² C总线上检测到停止条件后, 该位被清0。 |
| 9 | I2CMRxFO | 主机接收FIFO溢出。 接收FIFO已满后, 又有一个字节的数据写入FIFO时, 该位被置1。 在其它条件下, 该位被清0。 |
| 8 | I2CMTC | I ² C发送完成状态位。 当主机与从机之间完成一次传输后, 该位被置1。如果I2CMCON寄存器的I2CMCENI位被置1, 当该位的值为1时, 产生中断。 清除该中断源。 |
| 7 | I2CMND | I ² C主机非应答数据位。 在执行数据写传输过程中, 当主机接收到一个非应答条件后, 该位被置1。如果I2CMCON寄存器的I2CNACKENI位被置1, 当该位的值为1时, 产生中断。 在其它条件下, 该位被清0。 |
| 6 | I2CMBUSY | I ² C主机忙碌状态位。 当主机忙于处理事务时, 该位被置1。 当主机处于就绪状态或者当另一主机取得了总线控制权时, 该位被清0。 |
| 5 | I2CAL | I ² C仲裁失效状态位。 当I ² C主机未获得对I ² C总线的控制权时, 该位被置1。如果I2CMCON的I2CALENI位被置1, 当该位的值为1时, 产生中断。 在其它条件下, 该位被清0。 |
| 4 | I2CMNA | I ² C主机非应答地址位。 当主机接收到一个与地址对应的非应答条件后, 该位被置1。如果I2CMCON寄存器的I2CNACKENI位被置1, 当该位的值为1时, 产生中断。 在其它条件下, 该位被清0。 |
| 3 | I2CMRXQ | I ² C主机接收请求位。 当数据进入接收FIFO后, 该位被置1。如果I2CMCON的I2CMRENI位被置1, 则产生中断。 在其它条件下, 该位被清0。 |
| 2 | I2CMTXQ | I ² C主机发送请求位。 如发送FIFO为空或仅包含一个字节且主机已经传输一个地址并执行了写操作, 则该位置1。如果I2CMCON寄存器的I2CMTENI位被置1, 当该位被置1时, 产生中断。 在其它条件下, 该位被清0。 |
| 1:0 | I2CMTFSTA | I ² C主机发送FIFO状态位。 [00] = I ² C主机发送FIFO为空。 [01] = 在主机发送FIFO中含有1个字节的数据。 [10] = 在主机发送FIFO中含有1个字节的数据。 [11] = I ² C主机发送FIFO为空。 |

I2CMRX(I²C主机接收寄存器)

名称: I2CMRX
 地址: 0xFFFF0908
 默认值: 0x00
 访问类型: 只读
 功能: 该8位寄存器是I²C主机接收寄存器。

I2CMTX(I²C主机发送寄存器)

名称: I2CMTX
 地址: 0xFFFF090C
 默认值: 0x00
 访问类型: 只写
 功能: 该8位寄存器是I²C主机发送寄存器。

I2CMCNT0(I²C主机读取计数寄存器)

名称: I2CMCNT0
 地址: 0xFFFF0910
 默认值: 0x0000
 访问类型: 读、写
 功能: 该16位寄存器用于保存主机需要从从机中读取的字节数。

表99. I2CMCNT0 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|----------|--|
| 15:9 | | 保留。 |
| 8 | I2CRECNT | 当需要从从机处读取的字节数多于256时, 该位置1。 当需要从从机处读取的字节数小于等于256时, 该位清0。 |
| 7:0 | I2CRCNT | 这8位所保存的数值为需要从从机处读取的字节数减1的结果。如果只需要读取1个字节, 则将这些位清0。 |

I2CMCNT1(I²C主机当前读取计数寄存器)

名称: I2CMCNT1
 地址: 0xFFFF0914
 默认值: 0x00
 访问类型: 只读
 功能: 该8位寄存器用于保存在一次读序列中主机从从机处接收到的字节数。

I2CADR0(I²C地址0寄存器)

名称: I2CADR0
 地址: 0xFFFF0918
 默认值: 0x00
 访问类型: 读、写
 功能: 当主机开始与从机进行通信后, 该8位寄存器用于保存7位从机地址和读/写位。

表100. 7位地址模式下的I2CADR0 寄存器

| 位 | 名称 | 描述 |
|-----|--------|--------------------------------------|
| 7:1 | I2CADR | 这些位包含目标从机的7位地址。 位0为读/写位。 |
| 0 | R/W | 当该位置1时, 表示需要读指令。 当该位清零时, 表示需要写序列。 |

表101. 10位地址模式下的I2CADR0 寄存器

| 位 | 名称 | 描述 |
|-----|---------|---|
| 7:3 | | 在10位地址模式下, 这些位的值必须为[11110b]。 |
| 2:1 | I2CMADR | 在10位寻址模式下, 这些位包含ADDR[9:8]。 |
| 0 | R/W | 读/写位。 当该位置1时, 表示需要读指令。 当该位清零时, 表示需要写序列。 |

ADuC7060/ADuC7061

I2CADR1(I²C地址1寄存器)

名称: I2CADR1
地址: 0xFFFF091C
默认值: 0x00
访问类型: 读、写
功能: 该8位寄存器只可用于10位寻址模式。它包含地址的最低有效字节。

表102. 10位地址模式下的I2CADR1 寄存器

| 位 | 名称 | 描述 |
|-----|---------|----------------------------|
| 7:0 | I2CLADR | 在10位寻址模式下, 这些位包含ADDR[7:0]。 |

I2CDIV(I²C主机时钟控制寄存器)

名称: I2CDIV
地址: 0xFFFF0924
默认值: 0x1F1F
访问类型: 读、写
功能: 该寄存器用于控制SCL引脚上由主机产生的I²C时钟的频率。更多详情请参见“串行时钟生成”部分。

表103. I2CDIV 寄存器位功能描述

| 位 | 名称 | 描述 |
|------|------|-----------------------|
| 15:8 | DIVH | 这些位用于控制SCL高电平周期的持续时间。 |
| 7:0 | DIVL | 这些位用于控制SCL低电平周期的持续时间。 |

I²C从机寄存器

I2CSCON(I²C从机控制寄存器)

名称: I2CSCON
地址: 0xFFFF0928
默认值: 0x0000
访问类型: 读、写
功能: 该16位寄存器用于在从机模式下配置I²C外设。

表104.I2CSCON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|-----------|---|
| 15:11 | | 保留位。 |
| 10 | I2CSTXENI | 从机发送中断使能位。 将该位置1, 则当从机发送一个字节后, 产生中断。 清除该中断源。 |
| 9 | I2CSRXENI | 从机接收中断使能位。 将该位置1, 则当从机接收到数据后, 产生中断。 清除该中断源。 |
| 8 | I2CSSENI | I ² C停止条件检测中断使能位。 将该位置1, 则当从I ² C总线上检测到停止条件时, 产生中断。 清除该中断源。 |
| 7 | I2CNACKEN | I ² C非应答使能位。 将该位置1, 则对传输序列中的下一字节不作出应答。 将该位清0, 可允许硬件对应答/非应答序列进行控制。 |
| 6 | | 保留。应在此位写入0值。 |
| 5 | I2CSETEN | I ² C发送前中断使能位。 将该位置1, 则允许在读位发送的过程中, 在SCL正向沿之后立刻产生发送请求中断。 将该位置1, 则允许在读位发送的过程中, 在SCL负向沿之后立刻产生发送请求中断。 |
| 4 | I2CGCCLR | I ² C广播状态与ID清除位。 向该位写入1, 可清除I2CSSTA寄存器中的呼叫状态与ID位。 在其它状况下, 该位被清0。 |
| 3 | I2CHGCEN | 硬件广播使能位。当该位和位2置1时, 如果已接收到一个广播信号(地址为0x00)和1字节数据, 器件将对接收寄存器中的数据和I2CALT中的数据进行比较。如果数据匹配, 器件接收到一个硬件广播。这个功能主要用于当器件需要紧急呼叫一个主机而又不知道呼叫哪一个时, 这是“对可能相关的主机”的呼叫。ADuC7060/ADuC7061可监视这些地址。故器件需将自己的地址嵌入到消息中。所有的主机都会侦听这些消息, 然后其中合适的主机会同这个器件连接进行需要的操作。根据2000年1月的I ² C总线规范, I2CALT寄存器的LSB应该始终写入1。 |
| 2 | I2CGCEN | 广播使能位。将该位置1, 可以让从机为I ² C广播发送有效应答, 写地址0x00。然后器件将识别一个数据位。如果器件接收到的数据是0x06, 即由硬件复位和对从机地址的可编程部分进行写操作, 那么, 根据2000年1月的I ² C总线规范, 此时I ² C接口复位。这个命令可用于复位整个I ² C系统。如果接收到的数据为0x04, 即由硬件对从机地址可编程部分进行写操作, 则当产生任一广播后, 广播中断状态位被置位。在复位后, 用户必须通过重新对器件地址进行编程来进行恰当的操作。 |
| 1 | ADR10EN | I ² C 10位地址模式。 将该位置1, 以启用10位地址模式。 将该位清0, 以启用正常地址模式。 |
| 0 | I2CSEN | I ² C从机使能位。 用户将该位置1, 使能I ² C从机模式。 清0, 禁用I ² C从机模式。 |

ADuC7060/ADuC7061

I2CSSTA(I²C从机状态寄存器)

名称: I2CSSTA
 地址: 0xFFFF092C
 默认值: 0x0000
 访问类型: 读、写

该16位寄存器是从机模式下的I²C状态寄存器。

表105. I2CSSTA 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|--------------|---|
| 15 | | 保留位。 |
| 14 | I2CSTA | 当检测到匹配地址后面的起始条件并接收了起始字节(0x01)后, 或者当产生了一个广播且接收到广播的代码0x00后, 该位被置1。 当接收到停止条件后, 该位清0。 |
| 13 | I2CREPS | 检测到重复起始条件时, 将该位置1。 接收到停止条件后, 该位清0。I2CSSTA寄存器度将该位也清0。 |
| 12:11 | I2CID[1:0] | I ² C地址匹配寄存器。这些位用于表示哪个I2CIDx寄存器与接收到的地址相匹配。 [00] = 接收到的地址与I2CID0相匹配。 [01] = 接收到的地址与I2CID1相匹配。 [10] = 接收到的地址与I2CID2相匹配。 [11] = 接收到的地址与I2CID3相匹配。 |
| 10 | I2CSS | 起始检测位后的I ² C停止条件。 检测到起始位和匹配地址后, 如检测到停止位, 则该位置1。如果I2CSCON的I2CSSENI位置1, 则产生中断。 读该寄存器, 该位清0。 |
| 9:8 | I2CGCID[1:0] | I ² C广播ID位。 [00] = 未收到广播。 [01] = 广播复位和程序地址。 [10] = 通用程序地址。 [11] = 广播匹配可供选择的ID。 注意: 无法通过广播复位命令来清除这些位。 通过向I2CSCON的I2CGCCLR位写入1, 可清除这些位。 |
| 7 | I2CGC | I ² C广播状态位。 从机接收到任何类型的广播命令后, 该位置1。从机接收到复位命令后, 寄存器返回各自的默认状态。从机接收到硬件广播命令后, 接收FIFO将保存命令的第2个字节, 它可以与I2CALT寄存器的值进行比较。 通过向I2CSCON的I2CGCCLR位写入1, 可将该位清0。 |
| 6 | I2CSBUSY | I ² C从机忙碌状态位。 从机接收到起始条件后, 该位置1。 发生下列情形之一, 硬件会自动清除该位: 如接收到的地址与I2CIDx寄存器的内容不匹配, 从机收到停止条件, 或重复起始地址与I2CIDx寄存器的内容不匹配。 |
| 5 | I2CSNA | I ² C从机非应答数据位。 从机发出非应答信号以响应总线地址时, 该位被置1。在下列条件下, 该位被断言: 从机因发送FIFO内无数据而返回一非应答信号, 或者I2CSCON寄存器的I2CNACKEN位被置位。 在其它条件下, 该位被清0。 |
| 4 | I2CSRxFO | 从机接收FIFO溢出。 接收FIFO已满后, 又有一个字节的数据写入FIFO时, 该位被置1。 在其它条件下, 该位被清0。 |
| 3 | I2CSRXQ | I ² C从机接收请求位。 从机的接收FIFO不为空时, 该位被置1。I2CSCON寄存器的I2CSRXENI位置1时, 将I2CSRXQ置1, 将产生中断。 通过对接收FIFO执行读操作或清除操作, 该位清0。 |

| 位 | 名称 | 描述 |
|---|----------|--|
| 2 | I2CSTXQ | I ² C从机发送请求位。 从机接收到一个匹配的地址并执行读操作后，该位被置1。如果I2CSCON寄存器的I2CSETEN位清0，则在读位发送的过程中，SCL负向沿之后，I2CSTXQ的值立即被置1。如果I2CSCON寄存器的I2CSETEN位置1，则在读位发送的过程中，SCL正向沿之后，I2CSTXQ的值立即被置1。I2CSCON寄存器的I2CSTXEN位置1时，将I2CSTXQ置1，将产生中断。 在其它条件下，该位被清0。 |
| 1 | I2CSTFE | I ² C从机FIFO下溢状态位。 主机要求从机发送数据时，如果发送FIFO为空，则该位变为高电平。在读位操作期间，在SCL上升沿对该位进行断言。 在其它条件下，该位被清0。 |
| 0 | I2CETSTA | I ² C从机传输前FIFO状态位。 如果I2CSCON寄存器的I2CSETEN位清0，则当从机发送FIFO为空时，I2CETSTA变为高电平。如果I2CSCON寄存器的I2CSETEN位置1，则在写位发送的过程中，该位在SCL正向沿出现后立即变为高电平。 在一次传输过程中，该位只能被断言一次。 对该位执行读操作后，该位被清0。 |

I2CSRX(I²C从机接收寄存器)

名称: I2CSRX
地址: 0xFFFF0930
默认值: 0x00
访问类型: 只读
功能: 该8位寄存器是I²C从机接收寄存器。

I2CSTX(I²C从机发送寄存器)

名称: I2CSTX
地址: 0xFFFF0934
默认值: 0x00
访问类型: 只写
功能: 该8位寄存器是I²C从机发送寄存器。

I2CALT(I²C硬件广播识别寄存器)

名称: I2CALT
地址: 0xFFFF0938
默认值: 0x00
访问类型: 读、写
功能: I2CSCON寄存器的位3置1时，8位的I2CALT寄存器可用于识别硬件广播。主机无法为从机生成地址，相反，从机必须为主机生成地址时，该寄存器便可以发挥它的作用了。

I2CIDx(I²C从机ID寄存器)

名称: I2CIDx
地址: 0xFFFF093C = I2CID0
0xFFFF0940 = I2CID1
0xFFFF0944 = I2CID2
0xFFFF0948 = I2CID3
默认值: 0x00
访问类型: 读、写
功能: 通过编程，可在这些8位寄存器内编程从机的I²C总线ID。详情请参见对I²C总线地址部分的说明。

ADuC7060/ADuC7061

I²C公共寄存器

I2CFSTA(I²C FIFO状态寄存器)

| | |
|-------|----------------------------------|
| 名称: | I2CFSTA |
| 地址: | 0xFFFF094C |
| 默认值: | 0x0000 |
| 访问类型: | 读、写 |
| 功能: | 该16位寄存器用于存储在主、从机模式下接收/发送FIFO的状态。 |

表106. I2CFSTA 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|-----------|---|
| 15:10 | | 保留位。 |
| 9 | I2CFMTX | 将该位置1, 以清除主机发送FIFO。 |
| 8 | I2CFSTX | 将该位置1, 以清除从机发送FIFO。 |
| 7:6 | I2CMRXSTA | I ² C主机接收FIFO状态位。 [00] = FIFO为空。 [01] = FIFO字节写入。 [10] = FIFO中有1字节数据。 [11] = FIFO已满。 |
| 5:4 | I2CMTXSTA | I ² C主机发送FIFO状态位。 [00] = FIFO为空。 [01] = FIFO字节写入。 [10] = FIFO中有1字节数据。 [11] = FIFO已满。 |
| 3:2 | I2CSRSTA | I ² C从机接收FIFO状态位。 [00] = FIFO为空。 [01] = FIFO字节写入 [10] = FIFO中有1字节数据 [11] = FIFO已满 |
| 1:0 | I2CSTXSTA | I ² C从机发送FIFO状态位。 [00] = FIFO为空。 [01] = FIFO字节写入。 [10] = FIFO中有1字节数据。 [11] = FIFO已满。 |

串行外设接口

ADuC7060/ADuC7061片内集成了一个完整的硬件串行外设接口(SPI)。SPI是一个工业标准同步串行接口，允许同时双向传输8位数据(即，全双工)，最大比特率可达5.12 Mbps。

该SPI端口可配置为主机或从机操作，一般由4个引脚组成：MISO、MOSI、SCLK和 \overline{SS} 。

MISO(主机输入，从机输出)引脚

在主模式下，MISO引脚被配置为输入；在从模式下，配置为输出。主机上的MISO线路(数据输入)应与从机内的MISO线路(数据输出)相连。传输的数据均为以字节宽(8位)为单位的串行数据，最高有效位最先被传输。

MOSI(主机输出，从机输入)引脚

在主模式下，MISO引脚被配置为输出，在从模式下，配置为输入。主机上的MOSI线路(数据输出)应与从机内的MOSI线路(数据输入)相连。传输的数据均为以字节宽(8位)为单位的串行数据，最高有效位最先被传输。

SCLK(串行时钟输入/输出)引脚

主机串行时钟(SCL)用于在MOSI SCLK周期内同步发送和接收数据。所以，每隔8个SCLK周期，芯片发送/接收一个字节。在主机模式下，SCLK引脚配置成输出，而在从机模式下，配置成输入。

在主机模式下，时钟的极性和相位由SPICON寄存器控制，SPIDIV寄存器的值决定了比特率。比特率的计算公式如下：

$$f_{SERIAL\ CLOCK} = \frac{f_{UCLK}}{2 \times (1 + SPIDIV)}$$

SPI时钟的最高频率与时钟频率器位无关。

```
GP0CON0 = BIT0 + BIT4 + BIT8 + BIT12; //Select SPI/I2C alternative function for P0[0...3]
GP0KEY1 = 0x7; //Write to GP0KEY1
GP0CON1 &=~ BIT1; //Select SPI functionality for P0.0 to P0.3
GP0KEY2 = 0x13; //Write to GP0KEY2
```

在从机模式下，可对SPICON寄存器进行设置，以配置预期输入时钟的相位和极性。从机可以从外部主机处接收数据(速率可达5.12 Mbps)。

在主机或从机模式下，在SCLK信号的一个沿启动数据发送，在该信号的另一个沿启动数据采样。因此，从机时钟的极性和相位必须与主机的配置一致。

从机选择(P0.0/ \overline{SS})输入引脚

在SPI从机模式时，在P0.0/ \overline{SS} 引脚上断言 \overline{SS} ，可启动一次数据传输。该引脚为一个低电平有效信号。然后，SPI端口开始发送和接收8位数据，直到发送结束时为止，此时 \overline{SS} 无效。在从机模式下， \overline{SS} 总是输入。

在SPI主机模式下， \overline{SS} 是低电平有效输出信号。传输开始后，它自动断言；传输完成后，它自动无效。

SPI功能的外部引脚配置

ADuC7060/ADuC7061器件的SPI引脚由P0[0:3]引脚实现。

- P0.0/ \overline{SS} ：从机片选引脚。在从机模式下，该引脚作为输入引脚，必须通过主机将其驱动为低电平。在主机模式下，该引脚为输出引脚。当传输开始后，该引脚的电平为低电平，当传输完成后，该引脚的电平为高电平。
- P0.1/SCLK/SCL：SCLK引脚。
- P0.2/MISO：主机输入、从机输出(MISO)引脚。
- P0.3/MOSI/SDA：主机输出、从机输入(MOSI)引脚。

为了在SPI模式下对P0.0至P0.3进行配置，必须将GP0CON0寄存器的位0、4、8、12分别置1。GP0CON1寄存器的位1必须置1。注意：向GP0CON1内写入数据前，必须先将GP0KEY1寄存器的值设置为0x7。在对GP0CON1寄存器进行写操作之后，必须立刻将GP0KEY2寄存器的值设置为0x13。详见下列代码示例：

ADuC7060/ADuC7061

SPI寄存器

下列寄存器寄存器可用来控制SPI接口：SPISTA、SPIRX、SPITX、SPIDIV和SPICON。

SPI状态寄存器

SPISTA寄存器

名称： SPISTA

地址： 0xFFFF0A00

默认值： 0x00000000

访问类型： 只读

功能： 该32位寄存器用于存储主、从机模式下SPI接口的状态。

表107. SPISTA 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|----------------|--|
| 15:12 | | 保留位。 |
| 11 | SPIREX | SPI接收FIFO过剩字节存在。接收FIFO中的字节的个数超过由SPICON寄存器中的SPIMDE位规定的个数后，SPIREX被置1。 FIFO中的字节的个数不超过由SPICON寄存器中的SPIMDE位规定的个数时，该字节被清0。 |
| 10:8 | SPIRXFSTA[2:0] | SPI接收FIFO状态位。 [000] = 接收FIFO为空。 [001] = FIFO内有1个有效字节。 [010] = FIFO内有2个有效字节。 [011] = FIFO内有3个有效字节。 [100] = FIFO内有4个有效字节。 |
| 7 | SPIFOF | SPI接收FIFO溢出状态位。 接收FIFO已满，再一次向该FIFO内写入数据时，该位被置1。除非SPICON寄存器的SPIRFLH位置1，否则该位被置1将产生一个中断。 读取SPISTA寄存器的内容后，该位被清0。 |
| 6 | SPIRXIRQ | SPI接收IRQ状态位。 产生接收中断时，该位被置1。SPICON寄存器的SPITMDE位置1时，在接收到所需字节数后，SPIRXIRQ位被置1。 读取SPISTA寄存器的内容后，该位被清0。 |
| 5 | SPITXIRQ | SPI发送IRQ状态位。 产生发送中断时，该位被置1。SPICON寄存器的SPITMDE位置1时，在发送所需字节数后，SPITXIRQ位被置1。 读取SPISTA寄存器的内容后，该位被清0。 |
| 4 | SPITXUF | SPI发送FIFO下溢。 启动了一次发送操作且发送FIFO内没有有效数据时，该位被置1。除非SPICON寄存器的SPITFLH位置1，否则该位被置1将产生一个中断。 读取SPISTA寄存器的内容后，该位被清0。 |
| 3:1 | SPITXFSTA[2:0] | SPI发送FIFO状态位。 [000] = 发送FIFO为空。 [001] = FIFO内有1个有效字节。 [010] = FIFO内有2个有效字节。 [011] = FIFO内有3个有效字节。 [100] = FIFO内有4个有效字节。 |
| 0 | SPIISTA | SPI中断状态位。 SPI中断发生时，该位被置1。 读取SPISTA寄存器的内容后，该位被清0。 |

SPI接收寄存器**SPIRX寄存器**

名称: SPIRX

地址: 0xFFFF0A04

默认值: 0x00

访问类型: 只读

功能: 该8位寄存器是SPI接收寄存器。

SPI发送寄存器**SPITX寄存器**

名称: SPITX

地址: 0xFFFF0A08

默认值_y: 0x00

访问类型: 只写

功能: 该8位寄存器是SPI发送寄存器。

SPI波特率选择寄存器**SPIDIV寄存器**

名称: SPIDIV

地址: 0xFFFF0A0C

默认值: 0x1B

访问类型: 只写

功能: 该8位寄存器是SPI波特率选择寄存器。

表108. SPIDIV MMR位功能描述

| 位 | 描述 |
|-----|---|
| 7:6 | 保留 |
| 5:0 | SPI波特率设置: $f_{\text{SERIAL CLOCK}} = \frac{f_{\text{UCLK}}}{2 \times (1 + \text{SPIDIV})}$ |

SPI控制寄存器**SPICON寄存器**

名称: SPICON

地址: 0xFFFF0A10

默认值: 0x0000

访问类型: 读/写

功能: 该16位寄存器用于在主机和从机模式下配置SPI外设。

ADuC7060/ADuC7061

表109. SPICON 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|---------|---|
| 15:14 | SPIMDE | SPI IRQ模式位。在发送过程中发生发送/接收中断时，这些位被置1。 [00] = 发送完1个字节后，产生发送中断。FIFO接收到一个或一个以上字节后，产生接收中断。 [01] = 发送完2个字节后，产生发送中断。FIFO接收到两个或两个以上字节后，产生接收中断。 [10] = 发送完3个字节后，产生发送中断。FIFO接收到三个或三个以上字节后，产生接收中断。 [11] = 发送完4个字节后，产生发送中断。接收FIFO空间已满或有4个字节数据后，产生接收中断。 |
| 13 | SPITFLH | SPI发送FIFO清除使能位。 将该位置1，以清除发送FIFO。该位无法将其本身清0；需要一个单次清除操作时，应将该位置1。如果该位的值总保持为1，那么，发送0x00还是最后被发送的数值取决于SPIZEN位的值。该位为1时，无法对发送FIFO进行写操作。 将该位清0，可以禁用发送FIFO清除。 |
| 12 | SPIRFLH | SPI接收FIFO清除使能位。 将该位置1，以清除接收FIFO。该位无法将其本身清0；需要一个单次清除操作时，应将该位置1。该位被置1后，所有向接收FIFO写数据的操作将被忽略，且系统不产生中断。如果SPIRFLH = 1、SPITMDE = 0，对接收FIFO执行读操作可以启动一次数据传输。 将该位清0，可以禁用接收FIFO清除。 |
| 11 | SPICONT | 连续发送使能。 用户置1可以使能连续发送功能。在主机模式下，主机连续发送数据，直到发送寄存器内无有效数据为止。SS被置位后，在每一次8位串行传输期间，如发送寄存器不为空，SS可始终保持有效。 将该位清0，可禁用连续发送功能。 每一次发送8位串行数据。 如果SPITX寄存器中存在有效数据，那么在一个串行时钟周期后会重新开始发送数据。 |
| 10 | SPILP | 回送使能位。 将该位置1，以便将MISO连接到MOSI，用于测试软件。 将该位清0，以返回正常模式。 |
| 9 | SPIOEN | 从机MISO输出使能位。 将该位置1，以便让MISO在正常模式下工作。 将该位清0，可禁用MISO引脚上的输出驱动。该位被清0后，MISO引脚变为开漏极。 |
| 8 | SPIROW | SPIRX上溢覆盖使能。 用户置1，则新接收到的串行数据将覆盖接收寄存器中的有效数据。 用户清0，则新接收到的串行数据会被丢弃。 |
| 7 | SPIZEN | 发送FIFO为空时，SPI发送0。 将该位置1，则在发送FIFO无有效数据时，SPI发送0x00。 将该位清0，则在发送FIFO无有效数据时，SPI发送最后一次发送的数值。 |
| 6 | SPITMDE | SPI传输和中断模式。 用户置1，以启动数据传输，同时向SPITX寄存器写入数据。发送FIFO为空时，产生中断。 用户清0，以启动数据传输，同时从SPI寄存器中读取数据。接收FIFO为空时，产生中断。 |
| 5 | SPILF | LSB先发送使能位。 用户置1，先发送LSB。 用户清0，先发送MSB。 |
| 4 | SPIWOM | SPI线或模式使能位。 将该位置1，使能开漏极数据输出。数据输出引脚需要外部上拉电阻。 将该位清0，启动正常输出模式。 |
| 3 | SPICPO | 串行时钟极性模式位。 用户置1，则串行时钟空闲时高电平。 用户清0，则串行时钟空闲时低电平。 |
| 2 | SPICPH | 串行时钟相位模式位。 用户置1，串行时钟脉冲出现在每一个串行位发送的起始位置。 用户清0，串行时钟脉冲出现在每一个串行位发送的末尾。 |

| 位 | 名称 | 描述 |
|---|---------|--|
| 1 | SPI MEN | 主机模式使能位。 用户置1，使能主机模式。 用户清0，使能从机模式。 |
| 0 | SPI EN | SPI使能位。 用户置1，使能SPI。 用户清0，禁用SPI。 |

ADuC7060/ADuC7061

GPIO

ADuC7060/ADuC7061共有16个双向通用输入/输出(GPIO)引脚。一般情况下,通过用户代码设置,许多GPIO引脚可以实现多种功能。在默认的情况下,GPIO引脚在GPIO模式下工作。所有GPIO引脚都带有一个内部上拉电阻(驱动电流为1.6 mA)。

所有输入输出引脚的电压均为3.3 V,也就是说,GPIO支持的输入电压为3.3 V。

ADuC7060/ADuC7061进入节电模式后,GPIO引脚维持各自原来的状态。

GPIO引脚被分为3组端口总线。

表110列出所有GPIO引脚以及它们的备选功能。通过向GPxCON寄存器的某一位进行写操作,可以启动相应的GPIO引脚的备选功能。

表110.GPIO多功能引脚的描述

| 端口 | 通过GPxCON(包括GP0CON0)进行配置 | | |
|----|-------------------------|------------------|-----------------------|
| | 引脚名称 | 00 | 01 |
| 0 | P0.0/SS | GPIO | SS(SPI从机选择)。 |
| | P0.1/SCLK/SCL | GPIO | SCLK/SCL(串行时钟/SPI时钟)。 |
| | P0.2/MISO | GPIO | MISO(SPI-主机输入/从机输出)。 |
| | P0.3/MOSI/SDA | GPIO | MISO(SPI-主机输出/从机输入)。 |
| | P0.4/IRQ0/PWM1 | GPIO/IRQ0 | PWM1(PWM输出1)。 |
| | P0.5/CTS | GPIO | CTS(UART清除发送引脚)。 |
| | P0.6/RTS | GPIO | RTS(UART请求发送引脚)。 |
| 1 | P1.0/IRQ1/SIN/T0 | GPIO/IRQ1 | SIN(串行输入)。 |
| | P1.1/SOUT | GPIO | SOUT(串行输出)。 |
| | P1.2/SYNC | GPIO | PWM同步(PWM同步输入引脚)。 |
| | P1.3/TRIP | GPIO | PWM触发(PWM触发输入引脚)。 |
| | P1.4/PWM2 | GPIO | PWM2(PWM输出2)。 |
| | P1.5/PWM3 | GPIO | PWM3(PWM输出3)。 |
| | P1.6/PWM4 | GPIO | PWM4(PWM输出4)。 |
| 2 | P2.0/IRQ2/PWM0/EXTCLK | GPIO/IRQ2/EXTCLK | PWM0(PWM输出0)。 |
| | P2.1/IRQ3/PWM5 | GPIO/IRQ3 | PWM5(PWM输出5)。 |

GPxCON寄存器

GPxCON是端口x(x为: 0、1、2)的控制寄存器,它可以决定端口x中每一个引脚的功能。引脚功能如表112所示。

表111. GPxCON寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|---------|------------|------------|------|
| GP0CON0 | 0xFFFF0D00 | 0x00000000 | 读/写 |
| GP1CON | 0xFFFF0D04 | 0x00000000 | 读/写 |
| GP2CON | 0xFFFF0D08 | 0x00000000 | 读/写 |

表112. GPxCON 寄存器位功能描述

| 位 | 描述 |
|-------|---|
| 31:30 | 保留。 |
| 29:28 | 保留。 |
| 27:26 | 保留。 |
| 25:24 | 选择P0.6/RTS引脚和P1.6/PWM引脚的功能。 |
| 23:22 | 保留。 |
| 21:20 | 选择P0.5/CTS引脚和P1.5/PWM3引脚的功能。 |
| 19:18 | 保留。 |
| 17:16 | 选择P0.4/IRQ0/PWM1引脚和P1.4/PWM2引脚的功能。 |
| 15:14 | 保留。 |
| 13:12 | 选择P0.3/MOSI/SDA引脚和P1.3/TRIP引脚的功能。 |
| 11:10 | 保留。 |
| 9:8 | 选择P0.2/MISO引脚和P1.2/SYNC引脚的功能。 |
| 7:6 | 保留。 |
| 5:4 | 选择P0.1/SCLK/SCL、P1.1/SOUT和P2.1/IRQ3/PWM5引脚的功能。 |
| 3:2 | 保留。 |
| 1:0 | 选择P0.0/ \overline{SS} 、P1.0/IRQ1/SIN/T0和P2.0/IRQ2/PWM0/EXTCLK引脚的功能。 |

GPxPAR寄存器

通过对GPxPAR寄存器编程，可分别控制端口0、1、2的参数。注意：在对GPxDAT 寄存器执行写操作前，必须先对GPxPAR 寄存器执行写操作。注意：不可禁用P0.2引脚的内部上拉电阻。

表113. GPxDAT寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|--------|------------|------------|------|
| GP0DAT | 0xFFFF0D20 | 0x000000XX | R/W |
| GP1DAT | 0xFFFF0D30 | 0x000000XX | R/W |
| GP2DAT | 0xFFFF0D40 | 0x000000XX | R/W |

表114. GPxDAT 寄存器位功能描述

| 位 | 描述 |
|-------|---|
| 31:24 | 数据传输方向。 用户置1，可以将GPIO引脚配置为输出引脚。 用户清0，可以将GPIO引脚配置为输入引脚。 |
| 23:16 | 端口x数据输出。 |
| 15:8 | 反映复位时端口x引脚的状态(只读)。 |
| 7:0 | 端口x数据输入(只读)。 |

GPxSET寄存器

GPxSET为端口x的数据设置寄存器。

表115. GPxSET寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|--------|------------|------------|------|
| GP0SET | 0xFFFF0D24 | 0x000000XX | 写 |
| GP1SET | 0xFFFF0D34 | 0x000000XX | 写 |
| GP2SET | 0xFFFF0D44 | 0x000000XX | 写 |

表116. GPxSET 寄存器位功能描述

| 位 | 描述 |
|-------|---|
| 31:24 | 保留。 |
| 23:16 | 数据端口x设置位。 用户置1，则端口x的相应位被置位，同时，GPxDAT寄存器中的相应位也被置位。 用户清0不会影响数据输出。 |
| 15:0 | 保留。 |

GPxCLR寄存器

GPxCLR为端口x的数据清除寄存器。

表117. GPxCLR寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|--------|------------|------------|------|
| GP0CLR | 0xFFFF0D28 | 0x000000XX | W |
| GP1CLR | 0xFFFF0D38 | 0x000000XX | W |
| GP2CLR | 0xFFFF0D48 | 0x000000XX | W |

表118. GPxCLR 寄存器位功能描述

| 位 | 描述 |
|-------|--|
| 31:24 | 保留。 |
| 23:16 | 数据端口x清除位。 用户置1，则端口x中的相应位被清0，同时GPxDAT寄存器的相应位也被清0。 用户清0不会影响数据输出。 |
| 15:0 | 保留。 |

GPxPAR寄存器

通过对GPxPAR寄存器编程，可分别控制端口0、1、2的参数。注意：在对GPxDAT 寄存器执行写操作前，必须先对GPxPAR 寄存器执行写操作。注意：不可禁用P0.2引脚的内部上拉电阻。

表119. GPxPAR寄存器

| 名称 | 地址 | 默认值 | 访问类型 |
|--------|------------|------------|------|
| GP0PAR | 0xFFFF0D2C | 0x00000000 | R/W |
| GP1PAR | 0xFFFF0D3C | 0x00000000 | R/W |
| GP2PAR | 0xFFFF0D4C | 0x00000000 | R/W |

ADuC7060/ADuC7061

表120. GPxPAR 寄存器位功能描述

| 位 | 名称 | 描述 |
|-------|-----------|--|
| 31:15 | | 保留。 |
| 23:16 | GPL[7:0] | 通用输入输出端口引脚功能锁寄存器。 GPL[7:0] = 0: 端口处于正常工作状态; GPL[7:0] = 1: 如果每一个引脚的相应位均为1, 那么向GPxCON或GPxDAT的相应位写入数据的操作是无效的。 |
| 15:8 | GPDS[7:0] | 驱动能力配置。该位是可配置的。 GPDS[x] = 0: 最大源电流可达2 mA。 GPDS[x] = 1: 最大源电流可达4 mA。 |
| 7:0 | GPPD[7:0] | 端口x[7:0]上拉禁用位。 GPPD[x] = 0: 上拉电阻有效。 GPPD[x] = 1: 上拉电阻无效。 |

GP0CON1控制寄存器

GP0CON1写入值如下: GP0KEY1 = 0x7, GP0CON1 = 用户设定值, GP0KEY2 = 0x13。

名称: GP0CON1
地址: 0xFFFF0468
默认值: 0x00
访问类型: 读、写
功能: 该寄存器用于控制多功能GPIO引脚P0.0、P0.1、P0.2和P0.3的功能。

表121. GP0CON1写序列

| 名称 | 值 |
|---------|-------|
| GP0KEY1 | 0x7 |
| GP0CON1 | 用户设定值 |
| GP0KEY2 | 0x13 |

表122. GP0CON1 寄存器位功能描述

| 位 | 名称 | 描述 |
|-----|---------------|--|
| 7:2 | 保留 | 这些位总是清0。 |
| 1 | SPII2CS EL | 该位决定P0.0至P0.3在I ² C模式还是SPI模式下工作。注意: 为使该位能够正常工作, 必须将GP0CON1寄存器位0清0。 将该位清0, 则P0.0、P0.1、P0.2和P0.3在SPI模式下工作。 将该位置1, 则P0.0、P0.1、P0.2和P0.3在I ² C模式下工作。 |
| 0 | ADCSEL | 该位的默认值为0。 该位决定P0.0至P0.3作为GPIO引脚还是ADC输入引脚。 将该位置1, 则P0.0、P0.1、P0.2和P0.3作为ADC输入引脚。 将该位清0, 则P0.0、P0.1、P0.2和P0.3作为数字输入输出引脚。 该位的默认值为0。 |

名称: GP0KEY1
地址: 0xFFFF0464
默认值: 0xXXXX
访问类型: 只写
功能: 如需向GP0CON1寄存器写入数据, 必须首先通过指令向该寄存器内写入0x07。

名称: GP0KEY2
地址: 0xFFFF046C
默认值: 0xXXXX
访问类型: 只写
功能: 如需向GP0CON1寄存器写入数据, 必须首先通过指令向该寄存器内写入0x13。

硬件设计考虑

电源

ADuC7060/ADuC7061工作电压范围为2.375 V至2.625 V。模拟电源引脚和数字电源引脚(分别对应AVDD和DVDD)是分离的,因此,AVDD不受系统DVDD线路上经常出现的干扰数字信号的影响。在这种模式下,器件可以在分离电源下工作;也就是说,各个电源的电压可以是不同的。例如,系统的DVDD工作电压为2.6 V,而AVDD电压为2.5 V;反之亦然。图28是一个典型的分离电源配置。

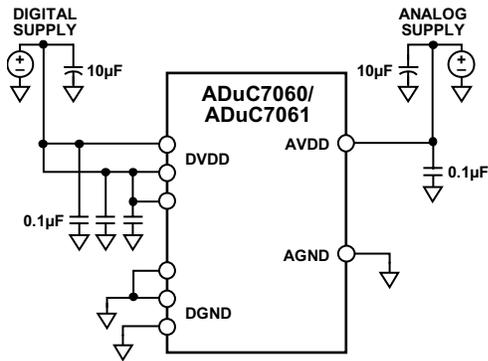


图28. 外部双电源连接

除了使用两个分离的电源以外,用户还可以通过在AVDD和DVDD之间串联一个小电阻和/或磁珠来降低AVDD的噪声,然后将AVDD对地单独去耦。图29所示的就是用这种方法进行设计的一个示例。使用这种方法,其它模拟电路(例如:运算放大器、基准电压源)也可以通过AVDD供电。

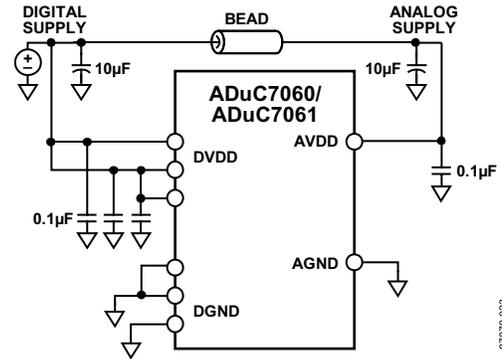


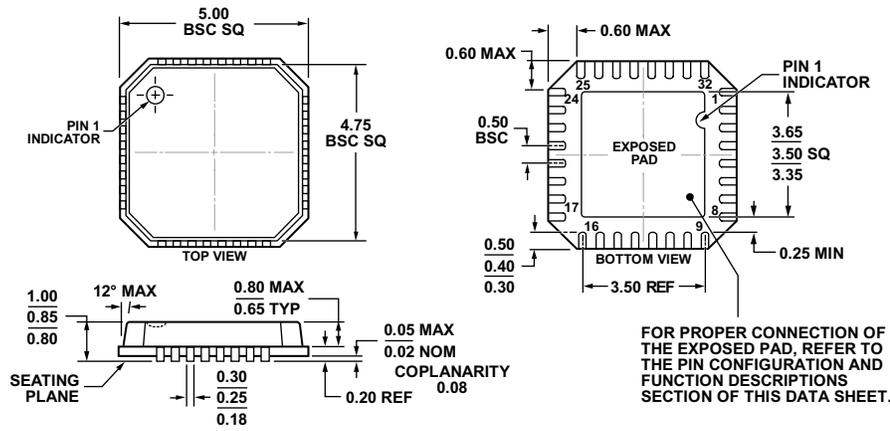
图29. 外部单电源连接

注意:在图28和图29中,在DVDD处有一个大容量值($10\ \mu\text{F}$)储能电容,在AVDD处有一个 $10\ \mu\text{F}$ 的电容。此外,在芯片的每一个AVDD和DVDD引脚都连接了一个小容值($0.1\ \mu\text{F}$)电容。在标准的设计中,必须确保包括所有这些电容且电容量越小的电容越接近AVDD引脚,布线长度也应尽量越短越好。这些电容的接地线端直接连接到地平面即可。

注意:无论在任何时候,ADuC7060/ADuC7061的模拟和数字接地引脚必须以同一系统接地基准点为基准。

最后还要注意一点,当DVDD的电压为1.8 V时,它必须在128 ms内斜升至2.25 V。这是内部上电复位电路要求必备的条件。

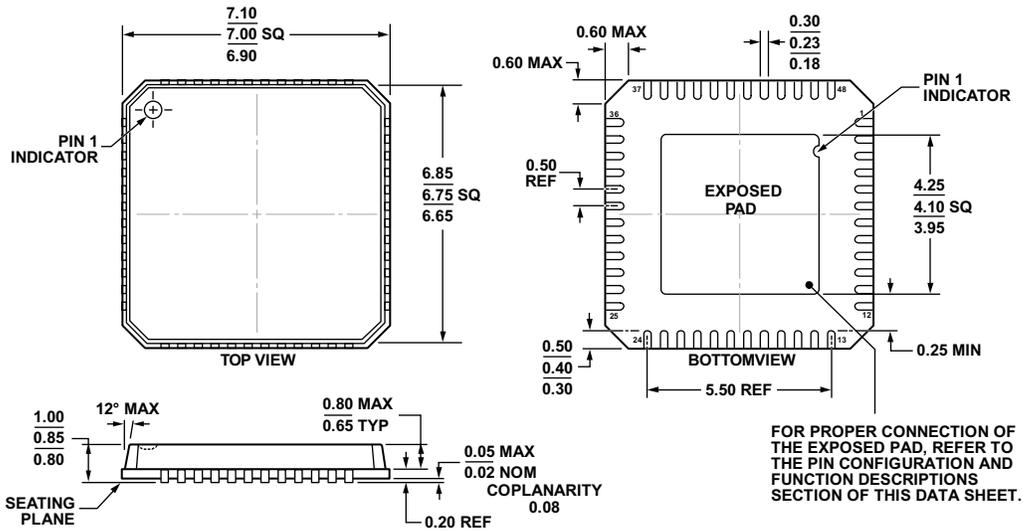
外形尺寸



COMPLIANT TO JEDEC STANDARDS MO-220-VHHD-2

图30. 32引脚LFCSP_VQ封装
5 mm × 5 mm, 超薄体
(CP-32-4)
尺寸单位: mm

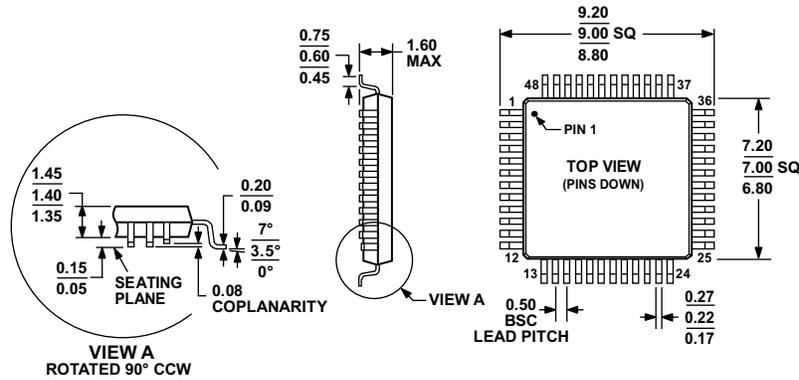
04-13-2012-A



COMPLIANT TO JEDEC STANDARDS MO-220-VKGD-2

图31. 48引脚LFCSP_VQ封装,
7 mm x 7 mm, 超薄体
(CP-48-3)
尺寸单位: mm

06-06-2012-A



COMPLIANT TO JEDEC STANDARDS MS-026-BBC

图32. 48引脚LQFP封装 (ST-48)

尺寸单位: mm

061706-A

订购指南

| 型号 ¹ | 温度范围 | 封装描述 | 封装选项 | 订购数量 |
|-------------------|--------------|---|---------|-------|
| ADuC7060BCPZ32 | -40°C至+125°C | 48引脚 LFCSP_VQ | CP-48-3 | |
| ADuC7060BCPZ32-RL | -40°C至+125°C | 48引脚 LFCSP_VQ | CP-48-3 | 2,500 |
| ADuC7060BSTZ32 | -40°C至+125°C | 48引脚 LQFP | ST-48 | |
| ADuC7060BSTZ32-RL | -40°C至+125°C | 48引脚 LQFP | ST-48 | 2,000 |
| ADuC7061BCPZ32 | -40°C至+125°C | 32引脚 LFCSP_VQ | CP-32-4 | |
| ADuC7061BCPZ32-RL | -40°C至+125°C | 32引脚 LFCSP_VQ | CP-32-4 | 5,000 |
| EVAL-ADuC7060QSPZ | | ADuC7060 Quick Start Plus开发系统 | | |
| EVAL-ADuC7061MKZ | | ADuC7061 Quick Start 评估系统 | | |

¹ Z = 符合RoHS标准的兼容器件。

注释