

### 产品特性

#### 模拟输入/输出

- 9通道、12位、1 MSPS ADC
  - 2个差分对，内置输入PGA
  - 7路通用输入(差分或单端)
- 全差分模式和单端模式
- 模拟输入范围：0 V至 $V_{REF}$
- 5个低噪声电流数模转换器(IDAC)
  - 250 mA、200 mA、45 mA、80 mA、20 mA
- 4个12位电压输出DAC
- 片内基准电压
- 片内温度传感器

#### 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

#### 时钟选项

- 修正的片内振荡器( $\pm 3\%$ )
- 外部时钟晶体
- 可达41.78 MHz的外部时钟源
- 具有可编程分频器的41.78 MHz锁相环

#### 存储器

- 126 kB Flash/EE存储器，8 kB SRAM
- 在线下载，基于JTAG调试

#### 软件触发在线重新编程能力

#### 片内外设

- UART、2个I<sup>2</sup>C和SPI串行I/O端口
- 32引脚GPIO端口
- 4个通用定时器
- 唤醒和看门狗定时器(WDT)
- 电源监控器
- 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

#### 电源

- 额定工作电压：3 V
- 主动模式：11 mA(5 MHz)，40 mA(41.78 MHz)

#### 封装和温度范围

- 7 mm x 7 mm 108引脚CSP\_BGA封装
- 额定工作温度范围：-10°C至+95°C

#### 工具

- 低成本QuickStart开发系统
- 完全第三方支持

#### 应用

- 光模块——可调激光

### 功能框图

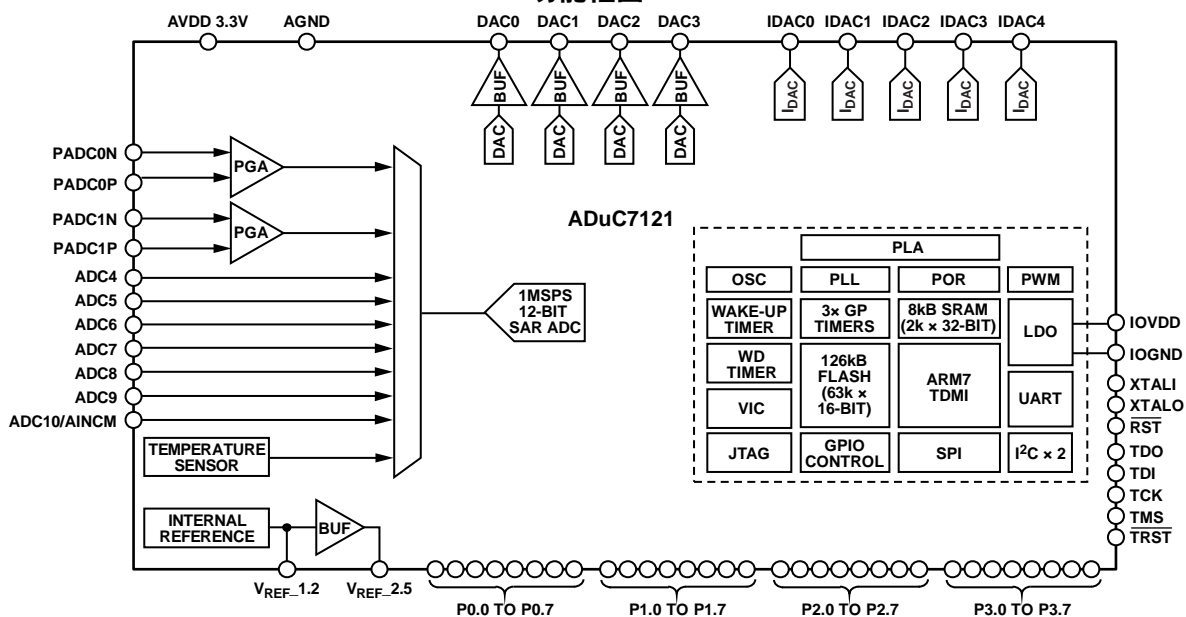


图1.

Rev. C

#### 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 ©2011–2014 Analog Devices, Inc. All rights reserved.  
Technical Support [www.analog.com](http://www.analog.com)

AD中文版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，AD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AD提供的最新英文版数据手册。

## 目录

产品特性 .....	1	其他模拟外设 .....	41
应用 .....	1	数模转换器 .....	41
功能框图 .....	1	LDO(低压差稳压器) .....	43
修订历史 .....	3	电流输出DAC (IDAC) .....	43
概述 .....	4	IDAC寄存器 .....	45
技术规格 .....	5	振荡器和锁相环—电源控制 .....	46
时序规格 .....	10	数字外设 .....	49
绝对最大额定值 .....	15	PWM概述 .....	49
ESD警告 .....	15	PWM转换开始控制 .....	51
引脚配置和功能描述 .....	16	通用输入/输出 .....	52
术语 .....	20	UART串行接口 .....	57
ADC规格 .....	20	波特率生成 .....	57
DAC技术规格 .....	20	UART寄存器定义 .....	57
ARM7TDMI内核概览 .....	21	I <sup>2</sup> C外设 .....	62
Thumb模式(T) .....	21	串行时钟生成 .....	62
长乘(M) .....	21	I <sup>2</sup> C总线地址 .....	62
EmbeddedICE (I) .....	21	I <sup>2</sup> C寄存器 .....	63
异常 .....	21	I <sup>2</sup> C公共寄存器 .....	71
ARM寄存器 .....	22	串行外设接口 .....	72
中断延迟时间 .....	22	SPI MISO(主机输入、从机输出)引脚 .....	72
存储器结构 .....	23	SPI MOSI(主机输出、从机输入)引脚 .....	72
存储器访问 .....	23	SPICLK(串行时钟输入/输出)引脚 .....	72
Flash/EE存储器 .....	23	SPI片选输入引脚 .....	72
SRAM .....	23	SPI功能的外部引脚配置 .....	72
存储器映射寄存器 .....	23	SPI寄存器 .....	72
完整MMR列表 .....	24	可编程逻辑阵列(PLA) .....	75
ADC电路概览 .....	27	PLA寄存器接口 .....	76
ADC传递函数 .....	27	中断系统 .....	79
温度传感器 .....	29	正常中断请求(IRQ) .....	79
转换器操作 .....	31	快速中断请求(FIQ) .....	80
驱动模拟输入 .....	33	外部中断(IRQ0至IRQ5) .....	84
带隙基准电压源 .....	33	定时器 .....	86
电源监控器 .....	34	时：分：秒：1/128格式 .....	86
非易失性Flash/EE存储器 .....	35	定时器0—使用寿命定时器 .....	86
Flash/EE存储器概述 .....	35	定时器1—通用定时器 .....	88
Flash/EE存储器 .....	35	定时器2—唤醒定时器 .....	90
Flash/EE存储器安全性 .....	36	定时器3—看门狗定时器 .....	91
Flash/EE控制接口 .....	36	定时器4—通用定时器 .....	93
SRAM和FLASH/EE执行时间 .....	39	硬件设计考虑 .....	95
复位和重映射 .....	39	电源 .....	95

接地和电路板布局建议 .....	96
时钟振荡器 .....	97

外形尺寸 .....	98
订购指南 .....	98

## 修订历史

### 2014年11月 — 修订版B至修订版C

更改“Flash/EE存储器”部分 .....	23
更改“PADC0x/PADC1x引脚”部分 .....	28
更改“Flash/EE存储器”部分和 “串行下载(在线编程)”部分 .....	35
更改“Flash/EE存储器安全”部分 .....	36
更改表41、表42和FEE0STA默认值 .....	37
更改“I <sup>2</sup> C外设”部分 .....	62
更改表8 .....	63
更改表96 .....	67
更改表97 .....	68
增加“硬件设计考虑”部分 .....	95

### 2013年3月 — 修订版A至修订版B

更改表9 .....	19
更改表11 .....	24
更改“复位操作”部分 .....	40
增加“RSTCFGKEY0寄存器”和 “RSTCFGKEY1寄存器”部分 .....	40
增加表49和表50，重新排序 .....	40
$I_{REF} = 370.37 \mu\text{A}$ 更改为 $I_{REF} = 380 \mu\text{A}$ .....	44
更改图32 .....	44
更改表57 .....	45
更改表64 .....	46

### 2012年3月 — 修订版0至修订版A

IDAC2满量程输出更改为45 mA .....	通篇
IDAC3满量程输出更改为80 mA .....	通篇
“技术规格”部分表1增加缓冲 $V_{REF}$ 输出参数 .....	6
更改“技术规格”部分表1的IDAC3和 IDAC2满量程输出参数 .....	6
更改“Flash/EE存储器”部分 .....	35
更改“PADC0x/PADC1x引脚”部分 .....	28
新增图15；重新排序 .....	28
更改“温度传感器”部分 .....	29
更改“电流输出DAC (IDAC)”部分 .....	43
更改表58 .....	45
更改“外部中断(IRQ0至IRQ5)”部分标题 .....	84
增加“时:分:秒:1/128格式”部分 .....	86
新增表122；重新排序 .....	86
更改“定时器3控制寄存器”部分 .....	91
更改表130 .....	92

### 2011年1月 — 修订版0：初始版

## 概述

ADuC7121是一款完全集成的1 MSPS、12位数据采集系统，在单芯片内集成高性能多通道ADC、16位/32位MCU和Flash®/EE存储器。

ADC具有多达7路单端输入和两对额外的差分输入。这两对差分输入可通过可编程增益放大器(PGA)进行路由。ADC可以在单端或差分输入模式下工作。ADC输入电压范围为0 V至 $V_{REF}$ 。低漂移带隙基准电压源、温度传感器和电压比较器完善了ADC的外设设置。

ADuC7121提供5个电流输出数模转换器(DAC)。电流源(5个电流DAC)可在11位分辨率下提供低噪声和低漂移高端电流输出。5个IDAC具体如下：IDAC0的满量程(FS)输出为250 mA，IDAC1为200 mA，IDAC2为45 mA，IDAC3为80 mA，IDAC4为20 mA。

ADuC7121同时提供4个电压输出数模转换器(DAC)。通过编程可以将DAC输出范围设置为三种电压范围之一。

这些器件通过一个片内振荡器和锁相环(PLL)产生41.78 MHz的内部高频时钟信号(UCLK)。该时钟信号通过一个可编程时钟分频器进行中继，在其中产生MCU内核时钟工作频率。微控制器内核为ARM7TDMI®，它是一个16位/32位RISC机器，峰值性能最高可达41 MIPS。片内集成有8 kB SRAM和126 kB非易失性Flash/EE存储器。ARM7TDMI内核将所有存储器和寄存器视为一个线性阵列。

片内出厂固件支持通过I<sup>2</sup>C串行接口端口进行串行在线下载，并且支持通过JTAG接口进行非介入仿真。这些特性都集成在支持此MicroConverter®系列的低成本QuickStart™开发系统中。

该器件采用3.0 V至3.6 V电源供电，额定温度范围为-10°C至+95°C工业温度范围。IDAC采用独立的2 V输入电源供电。当工作频率为41.78 MHz时，典型的功耗为120 mW。ADuC7121采用108引脚芯片级球栅阵列封装[CSP\_BGA]。

## 技术规格

除非另有说明,  $AV_{DD} = IOV_{DD} = 3.0\text{ V}$ 至 $3.6\text{ V}$ ,  $PV_{DD} = 2.0\text{ V} \pm 5\%$ ,  $V_{REF} = 2.5\text{ V}$ 内部基准电压,  $f_{CORE} = 41.78\text{ MHz}$ ,  
 $T_A = -10^\circ\text{C}$ 至 $+95^\circ\text{C}$ 。

表1.

参数	最小值	典型值	最大值	单位	测试条件/注释
ADC通道规格					8采集时钟和 $f_{ADC}/2$
ADC上电时间		5		$\mu\text{s}$	
直流精度 <sup>1,2</sup>	12			位	
分辨率				位	
积分非线性		$\pm 0.6$	$\pm 2$	LSB	2.5 V内部基准电压, PADC0和PADC1通道 未经生产测试
微分非线性 <sup>3,4</sup>		$\pm 0.5$	$+1.4/-0.99$	LSB	2.5 V内部基准电压, 保证单调性
直流代码分布		1		LSB	ADC输入为直流电压
端点误差 <sup>5</sup>					内部无缓冲通道
失调误差					
IDACx通道除外的所有通道		$\pm 2$	$\pm 5$	LSB	
仅IDACx通道		1		满量程 百分比	
失调误差匹配		$\pm 1$		LSB	
增益误差		$\pm 2$	$\pm 5$	LSB	
增益误差匹配		$\pm 1$		LSB	
动态性能					$f_{IN} = 10\text{ kHz}$ 正弦波, $f_{SAMPLE} = 1\text{ MSPS}$ , 内部无缓冲通道 包括失真和噪声成分
信噪比(SNR)		69		dB	
总谐波失真(THD)		-78		dB	
峰值谐波或杂散噪声		-75		dB	
通道间串扰		-80		dB	相邻通道测量
模拟输入					
输入电压范围					
差模			$V_{CM}^6 \pm V_{REF}/2$	V	见表38
单端模式	0.15		0至 $V_{REF}$ $AV_{DD} - 1.5$	V	缓冲旁路 缓冲使能
漏电流		$\pm 0.2$	$\pm 1$	$\mu\text{A}$	
输入电容		20		pF	ADC采样期间, 缓冲旁路
		20		pF	ADC采样期间, 缓冲使能
PADC0x输入					28.3 k $\Omega$ 电阻, PGA增益 = 3, 采集时间 = 3.2 $\mu\text{s}$ , 伪差分模式
满量程输入范围	20		1000	$\mu\text{A}$	
PADC0x的输入漏电流 <sup>4</sup>		0.15	2	nA	
分辨率	11			位	0.1%精度、5 ppm外部电阻用于将电流转换为电压
增益误差 <sup>4</sup>			1	%	
增益漂移 <sup>4</sup>			50	ppm/ $^\circ\text{C}$	
失调 <sup>4</sup>		3	6	nA	不包括PGA失调
失调漂移 <sup>4</sup>		30	60	pA/ $^\circ\text{C}$	
PADC0x顺从电压范围	0.1		$AV_{DD} - 1.2$	V	

# ADuC7121

参数	最小值	典型值	最大值	单位	测试条件/注释
PADC1输入					53.5 kΩ电阻, PGA增益 = 3
满量程输入范围	10.6		700	μA	
PADC1x的输入漏电流 <sup>4</sup>		0.15	2	nA	
分辨率	11			位	0.1%精度、5 ppm外部电阻用于将电流转换为电压
增益误差 <sup>4</sup>			1	%	
增益漂移 <sup>4</sup>			50	ppm/°C	
失调 <sup>4</sup>		3	6	nA	不包括PGA失调
失调漂移 <sup>4</sup>		30	60	pA/°C	
PADC1x顺从电压范围	0.1		AV <sub>DD</sub> - 1.2	V	
片内基准电压					在V <sub>REF</sub> 和AGND之间连接0.47 μF电容
输出电压		2.5		V	
精度 <sup>7</sup>			±5	mV	T <sub>A</sub> = 25°C
基准电压温度系数 <sup>4</sup>		10	30	ppm/°C	
电源抑制比		61		dB	
输出阻抗		10		Ω	T <sub>A</sub> = 25°C
内部V <sub>REF</sub> 上电时间		1		ms	
缓冲V <sub>REF</sub> 输出					
BUF_VREFx		±80		mV	
外部基准输入					
输入电压范围	1.2		AV <sub>DD</sub>	V	
IDAC通道规格					
电压范围	0.4		1.6	V	输出顺从电压
电压范围, IDAC0	-0.2		+1.6 <sup>8</sup>	V	仅IDAC0通道, 0 V以下不保证线性度
基准电流发生器					
基准电流		0.38		mA	使用内部基准电压, 0.1% 5 ppm 3.16 kΩ外部电阻
温度系数		25		ppm/°C	使用内部基准电压
短路检测		1		mA	
过热关断		135		°C	结温
分辨率		11		位	保证单调性
满量程输出					
IDAC4		20		mA	
IDAC3		80		mA	
IDAC2		45		mA	
IDAC1		200		mA	
IDAC0		250		mA	
积分非线性		±2		LSB	11位模式
噪声电流		20		μA	RMS值, 20 Hz至10 MHz带宽
满量程误差		±3		%	V <sub>OUT</sub> = 1.6 V
满量程误差漂移		50		ppm/°C	内部V <sub>REF</sub> , 5 ppm外部电阻
零刻度误差					下拉电阻断开, V <sub>OUT</sub> = 0 V
IDAC4通道			±30	μA	
IDAC3通道			+42/-70	μA	
IDAC2通道			+70/-110	μA	
IDAC1通道			±240	μA	
IDAC0通道			±250	μA	输出范围0.4 V至1.6 V
			+580/-430	μA	输出范围-0.2 V至+1.6 V
建立时间		1		ms	至0.1%
信号带宽		20		kHz	

参数	最小值	典型值	最大值	单位	测试条件/注释
电压调整率					在电流DAC驱动满量程电流负载的条件下测量
IDAC4		10		$\mu\text{A}/\text{V}$	
IDAC3		22.5		$\mu\text{A}/\text{V}$	
IDAC2		40		$\mu\text{A}/\text{V}$	
IDAC1		100		$\mu\text{A}/\text{V}$	
IDAC0		750		$\mu\text{A}/\text{V}$	
负载调整率					在电流DAC驱动满量程电流负载的条件下测量
IDAC4		10		$\mu\text{A}/\text{V}$	
IDAC3		22.5		$\mu\text{A}/\text{V}$	
IDAC2		40		$\mu\text{A}/\text{V}$	
IDAC1		100		$\mu\text{A}/\text{V}$	
IDAC0		750		$\mu\text{A}/\text{V}$	
ACPSRR <sup>4</sup>	0.75%			满量程百分比/V	10 kHz, 各电流DAC每伏满量程百分比
	6%			满量程百分比/V	2.25 MHz, 各电流DAC每伏满量程百分比
下拉					
NMOS		100		mV	漏极40 mA
速度 <sup>4</sup>			10	$\mu\text{s}$	由PLA触发, 将引脚电压拉至原始值的10%
电压DAC (VDAC)通道					$R_L = 5 \text{ k}\Omega, C_L = 100 \text{ pF}$
直流精度 <sup>9</sup>					缓冲
分辨率		12		位	
相对精度		$\pm 2$		LSB	
微分非线性		$\pm 0.2$	$\pm 1$	LSB	保证单调性
计算失调误差		$\pm 2$		mV	2.5 V内部基准电压
实际失调误差		9		mV	在代码0下测量
增益误差 <sup>10</sup>		$\pm 0.15$	$\pm 0.8$	%	
增益误差失配		0.1		%	DAC0满量程的百分比
建立时间		10		$\mu\text{s}$	
PSRR <sup>4</sup>					缓冲
直流	-59	-61		dB	
1 kHz	-57			dB	
10 kHz	-47			dB	
100 kHz	-19			dB	
漂移					
失调漂移 <sup>4</sup>			10	$\mu\text{V}/^\circ\text{C}$	
增益误差漂移 <sup>4</sup>			10	$\mu\text{V}/^\circ\text{C}$	
短路电流		20		mA	
模拟输出					
输出范围	0.1		$V_{\text{REF}}/AV_{\text{DD}} - 0.1$		缓冲开启
DAC交流特性					
压摆率		2.49		$\text{V}/\mu\text{s}$	
输出电压建立时间		10		$\mu\text{s}$	
数模转换脉冲干扰		$\pm 20$		nV-sec	主进位1 LSB变化(DACxDAT寄存器中同时变化的最大位数)

# ADuC7121

参数	最小值	典型值	最大值	单位	测试条件/注释
温度传感器 <sup>11</sup>					用户校准后
25°C时电压输出		707		mV	
电压温度系数		-1.25		mV/°C	
精度		±3		°C	测量前MCU处于掉电或待机模式
电源监控器(PSM)					
IOV <sub>DD</sub> 跳变点选择		2.79		V	两个可选择跳变点
		3.07		V	
电源跳变点精度		±2.5		%	已选跳变点标称电压
上电复位		2.36		V	
看门狗定时器(WDT)					
超时时间	0		512	s	
FLASH/EE存储器					
耐久性 <sup>12</sup>	10,000			周期	
数据保持 <sup>13</sup>	20			年	T <sub>J</sub> = 85°C
数字输入					除XTALI和XTALO外的所有数字输入
逻辑1输入电流		±0.2	±1	μA	V <sub>IH</sub> = V <sub>DD</sub>
逻辑0输入电流		-40	-60	μA	V <sub>IL</sub> = 0 V; TDI除外
输入电容		10		pF	
逻辑输入 <sup>4</sup>					除 XTALI 外的所有数字输入
输入低电压V <sub>INL</sub> <sup>4</sup>			0.8	V	
输入高电压V <sub>INH</sub> <sup>4</sup>	2.0			V	
逻辑输出					除 XTALO 外的所有数字输出
输出高电压V <sub>OH</sub>	2.4			V	I <sub>SOURCE</sub> = 1.6 mA
输出低电压V <sub>OL</sub> <sup>14</sup>			0.4	V	I <sub>SINK</sub> = 1.6 mA
晶体输入(XTALI和XTALO)					
逻辑输入, 仅限XTALI					
输入低电压V <sub>INL</sub>		1.1		V	
输入高电压V <sub>INH</sub>		1.7		V	
XTALI输入电容		20		pF	
XTALO输出电容		20		pF	
内部振荡器		32.768		kHz	
			±3	%	
MCU时钟速率					
采用32 kHz内部振荡器		326		kHz	时钟分频器(CD) = 7
采用32 kHz外部晶体		41.78		MHz	CD = 0
使用外部时钟	0.05		41.78	MHz	T <sub>A</sub> = 95°C
启动时间					内核时钟(HCLK) = 41.78 MHz
上电时		70		ms	
从暂停/休眠模式		24		ns	CD = 0
		3.06		μs	CD = 7
从休眠模式		1.58		ms	
从停止模式		1.7		ms	
可编程逻辑阵列(PLA)					
引脚传输延迟		12		ns	从输入引脚到输出引脚
单元传输延迟		2.5		ns	
电源要求 <sup>15, 16</sup>					
电源电压范围					
AV <sub>DD</sub> 至AGND和IOV <sub>DD</sub> 至IOGND	3.0		3.6	V	
模拟电源电流					
AV <sub>DD</sub> 电流		200		μA	ADC处于空闲模式



参数	最小值	典型值	最大值	单位	测试条件/注释
数字电源电流					
IOV <sub>DD</sub> 正常模式下电流		7		mA	从Flash/EE执行代码 CD = 7
		11		mA	CD = 3
		30	40	mA	CD = 0(时钟频率41.78 MHz)
IOV <sub>DD</sub> 暂停模式下电流 <sup>4</sup>		25		mA	CD = 0(时钟频率41.78 MHz)
IOV <sub>DD</sub> 休眠模式下电流 <sup>4</sup>		100		μA	T <sub>A</sub> = 25°C
附加电源电流					
模数转换器		2.7		mA	@1 MSPS
IDAC		21		mA	所有电流DAC (IDAC)开启
DAC		250		μA	每VDAC
ESD测试					2.5 V基准电压, T <sub>A</sub> = 25°C
最大HBM通过电压			4	kV	
最大FICDM通过电压			0.5	kV	

<sup>1</sup> 在MicroConverter内核正常工作时, 保证所有ADC通道的技术规格。

<sup>2</sup> 适用于所有ADC输入通道。

<sup>3</sup> 使用ADC失调寄存器(ADCOF)和增益系数寄存器(ADCGN)中的出厂设定默认值进行测试。

<sup>4</sup> 未经生产测试, 但量产时的设计和/或特性数据可提供保证。

<sup>5</sup> 采用外部运算放大器AD845作为输入缓冲级, 用ADCOF和ADCGN寄存器中的出厂设定默认值进行测试, 如图24所示。基于外部ADC系统元件, 用户可能需要进行系统校准来消除外部端点误差并满足规格要求(参见ADC部分)。

<sup>6</sup> 输入信号可以任何直流共模电压(V<sub>CM</sub>)为中心, 但该值必须位于ADC规定输入电压范围内。

<sup>7</sup> V<sub>REF</sub>校准和调整在以下条件下进行: 内核以正常模式工作, CD=0, ADC开启, 电流DAC开启, 所有VDAC开启。在其他工作条件下, V<sub>REF</sub>精度可能有所差异。

<sup>8</sup> PVDD\_IDAC0焊盘电压必须比IDAC0焊盘电压大至少300 mV。这些电压通过ADC的PVDD0和IDAC0通道测量。只要这两个焊盘之间的电压差保持300 mV, IDAC0引脚便可上拉至1.7 V。这可能要求PVDD\_IDAC0用2.0 V以上的电压供电。不得超过PVDD\_IDACx额定最大值2.1 V。

<sup>9</sup> DAC的线性度是使用一个递减的数据范围(100到3995)计算出来的。

<sup>10</sup> DAC增益误差是使用一个递减的数据范围(100到内部2.5 V基准电压)计算出来的。

<sup>11</sup> 芯片温度。

<sup>12</sup> 耐久性是在分别在-40°C、+25°C、+85°C及+125°C时依据JEDEC 22标准方法A117来认定的。

<sup>13</sup> 根据JEDEC 22标准方法A117, 保持期限相当于85°C结温时的寿命。保持期限会随着结温递减。

<sup>14</sup> 测试是在最多8个I/O端口输出低电平时进行的。

<sup>15</sup> 电源功耗分别在正常、暂停和休眠模式下测试的, 这三种模式下的测试条件分别为: 正常模式供电电压为3.6 V、暂停模式供电电压为3.6 V、休眠模式供电电压为3.6 V。

<sup>16</sup> 在一个Flash/EE擦写周期中, IOV<sub>DD</sub>电源电流通常提高2 mA。

# ADuC7121

## 时序规格

表2. 快速模式下I<sup>2</sup>C时序(400 kHz)

参数	说明	从机		主机	单位
		最小值	最大值	典型值	
$t_L$	SCL低电平脉宽	200		1360	ns
$t_H$	SCL高电平脉宽	100		1140	ns
$t_{SHD}$	起始条件保持时间	300		251,350	ns
$t_{DSU}$	数据建立时间	100		740	ns
$t_{DHD}$	数据保持时间	0		400	ns
$t_{RSU}$	重复起始建立时间	100		12.51350	ns
$t_{PSU}$	停止条件的建立时间	100		400	ns
$t_{BUF}$	一个结束条件和起始条件之间的总线空闲时间	1.3			$\mu$ s
$t_R$	SCL和SDA的上升时间		300	200	ns
$t_F$	SCL和SDA的下降时间		300		ns
$t_{SUP}$	尖峰抑制脉宽		50		ns

表3. 标准模式下I<sup>2</sup>C时序(100 kHz)

参数	说明	从机		单位
		最小值	最大值	
$t_L$	SCLx低电平脉冲宽度	4.7		$\mu$ s
$t_H$	SCLx高电平脉冲宽度	4.0		ns
$t_{SHD}$	起始条件保持时间	4.0		$\mu$ s
$t_{DSU}$	数据建立时间	250		ns
$t_{DHD}$	数据保持时间	0	3.45	$\mu$ s
$t_{RSU}$	重复起始建立时间	4.7		$\mu$ s
$t_{PSU}$	停止条件的建立时间	4.0		$\mu$ s
$t_{BUF}$	一个结束条件和起始条件之间的总线空闲时间	4.7		$\mu$ s
$t_R$	SCLx和SDAx的上升时间		1	$\mu$ s
$t_F$	SCLx和SDAx的下降时间		300	ns

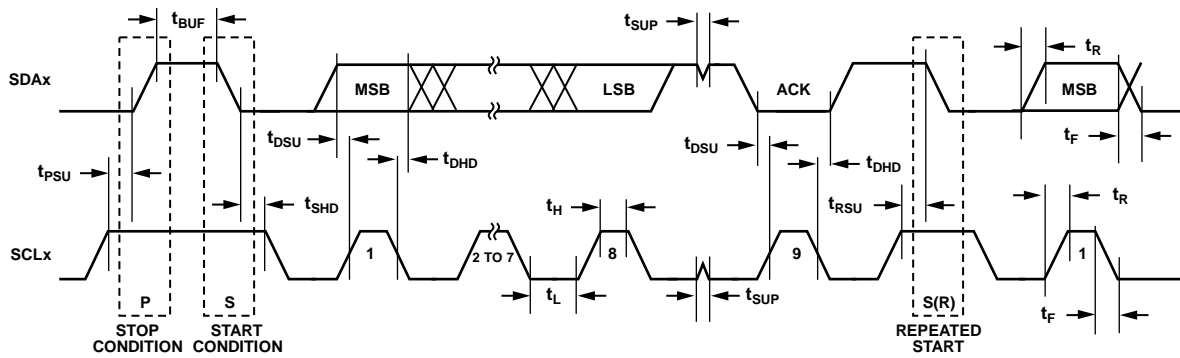


图2. I<sup>2</sup>C兼容接口时序

09492-002

表4. SPI主机定时(相位模式 = 1)

参数	说明	最小值	典型值	最大值	单位
$t_{SL}$	SPICLK低电平脉冲宽度		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{SH}$	SPICLK高电平脉冲宽度		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{DAV}$	SPICLK边沿之后数据输出有效时间			25	ns
$t_{DSU}$	SPICLK边沿之前数据输入建立时间 <sup>1</sup>	$1 \times t_{UCLK}$			ns
$t_{DHD}$	SPICLK边沿之后数据输入保持时间	$2 \times t_{UCLK}$			ns
$t_{DF}$	数据输出下降时间		5	12.5	ns
$t_{DR}$	数据输出上升时间		5	12.5	ns
$t_{SR}$	SPICLK上升时间		5	12.5	ns
$t_{SF}$	SPICLK下降时间		5	12.5	ns

<sup>1</sup>  $t_{UCLK} = 23.9$  ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟。

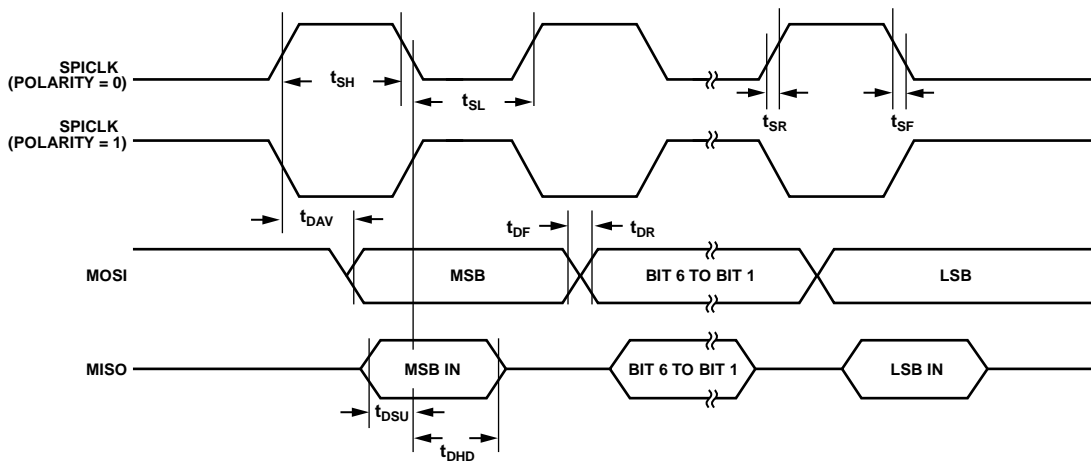


图3. SPI主机模式时序(相位模式 = 1)

09492-003

# ADuC7121

表5. SPI主机定时(相位模式 = 0)

参数	说明	最小值	典型值	最大值	单位
$t_{SL}$	SPICLK低电平脉冲宽度		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{SH}$	SPICLK高电平脉冲宽度		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{DAV}$	SPICLK边沿之后数据输出有效时间			25	ns
$t_{DOSU}$	SPICLK边沿之前数据输出建立时间			75	ns
$t_{DSU}$	SPICLK边沿之前数据输入建立时间 <sup>1</sup>	$1 \times t_{UCLK}$			ns
$t_{DHD}$	SPICLK边沿之后数据输入保持时间	$2 \times t_{UCLK}$			ns
$t_{DF}$	数据输出下降时间		5	12.5	ns
$t_{DR}$	数据输出上升时间		5	12.5	ns
$t_{SR}$	SPICLK上升时间		5	12.5	ns
$t_{SF}$	SPICLK下降时间		5	12.5	ns

<sup>1</sup>  $t_{UCLK} = 23.9$  ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟。

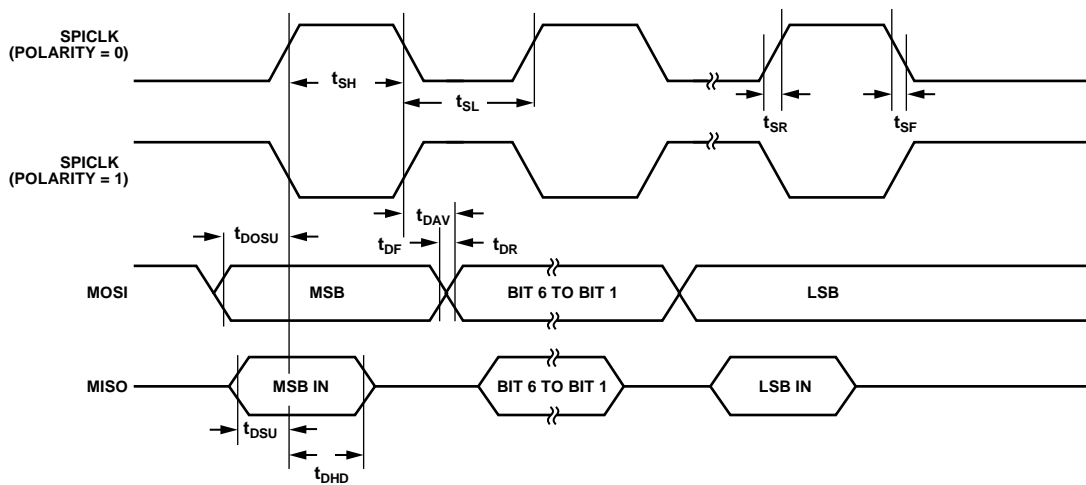


图4. SPI主机模式时序(相位模式 = 0)

09492-004

表6. SPI从机定时(相位模式 = 1)

参数	说明	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	$\overline{CS}$ 到SPICLK边沿 <sup>1</sup>	200			ns
$t_{SL}$	SPICLK低电平脉冲宽度 <sup>2</sup>		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{SH}$	SPICLK高电平脉冲宽度 <sup>2</sup>		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{DAV}$	SPICLK边沿之后数据输出有效时间			25	ns
$t_{DSU}$	SPICLK边沿之前数据输入建立时间	$1 \times t_{UCLK}$			ns
$t_{DHD}$	SPICLK边沿之后数据输入保持时间	$2 \times t_{UCLK}$			ns
$t_{DF}$	数据输出下降时间		5	12.5	ns
$t_{DR}$	数据输出上升时间		5	12.5	ns
$t_{SR}$	SPICLK上升时间		5	12.5	ns
$t_{SF}$	SPICLK下降时间		5	12.5	ns
$t_{SFS}$	SPICLK边沿之后 $\overline{CS}$ 高电平时间	0			ns

<sup>1</sup>  $\overline{CS}$ 是多功能引脚F3的 $\overline{CS}$ (SPI从机选择输入)功能。

<sup>2</sup>  $t_{UCLK} = 23.9$  ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟。

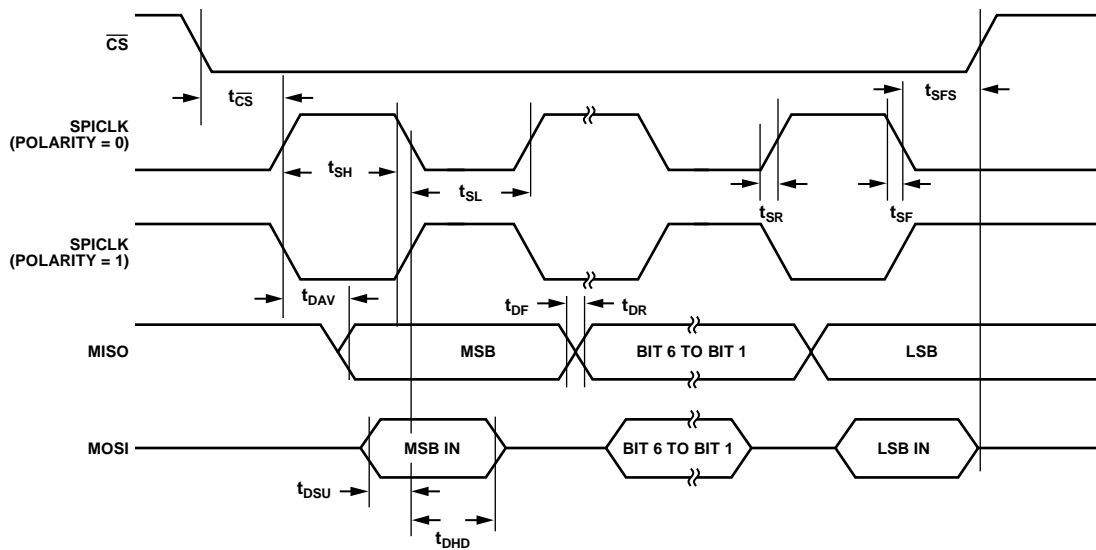


图5. SPI从机模式时序(相位模式 = 1)

09-492-005

# ADuC7121

表7. SPI从机定时(相位模式 = 0)

参数	说明	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	$\overline{CS}$ 到SPICLK边沿 <sup>1</sup>	200			ns
$t_{SL}$	SPICLK低电平脉冲宽度 <sup>2</sup>		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{SH}$	SPICLK高电平脉冲宽度 <sup>2</sup>		$(SPIDIV + 1) \times t_{UCLK}$		ns
$t_{DAV}$	SPICLK边沿之后数据输出有效时间			25	ns
$t_{DSU}$	SPICLK边沿之前数据输入建立时间 <sup>2</sup>	$1 \times t_{UCLK}$			ns
$t_{DHD}$	SPICLK边沿之后数据输入保持时间 <sup>2</sup>	$2 \times t_{UCLK}$			ns
$t_{DF}$	数据输出下降时间		5	12.5	ns
$t_{DR}$	数据输出上升时间		5	12.5	ns
$t_{SR}$	SPICLK上升时间		5	12.5	ns
$t_{SF}$	SPICLK下降时间		5	12.5	ns
$t_{DOCS}$	$\overline{CS}$ 边沿之后数据输出有效			25	ns
$t_{SFS}$	SPICLK边沿之后 $\overline{CS}$ 高电平时间	0			ns

<sup>1</sup>  $\overline{CS}$ 是多功能引脚F3的 $\overline{CS}$ (SPI从机选择输入)功能。

<sup>2</sup>  $t_{UCLK} = 23.9$  ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟。

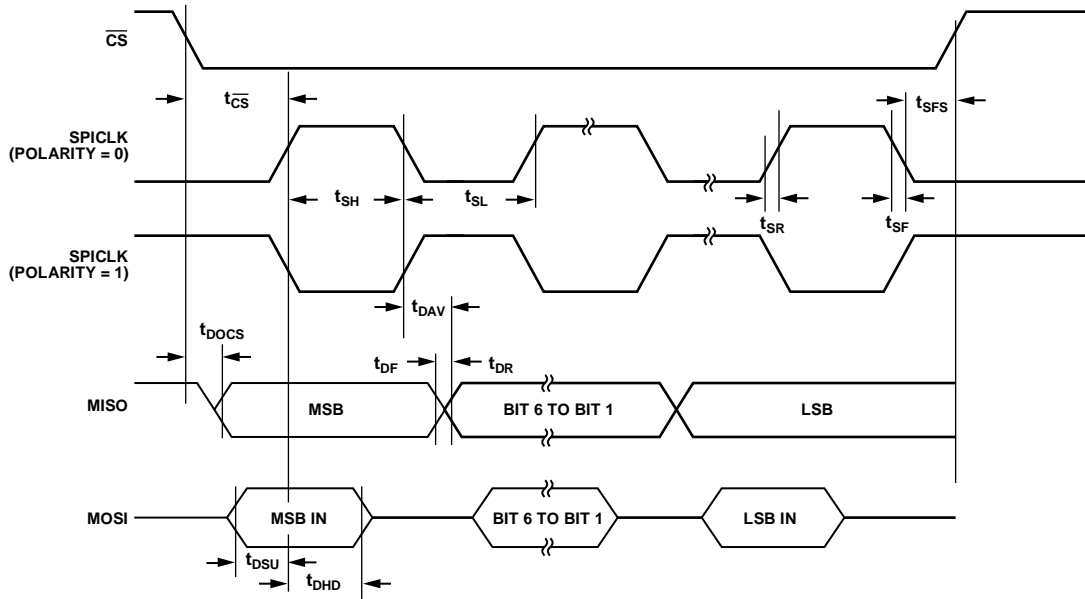


图6. SPI从机模式时序(相位模式 = 0)

09492-006

## 绝对最大额定值

除非另有说明，AGND = 0 V， $T_A = 25^\circ\text{C}$ 。

表8.

参数	额定值
$AV_{DD}$ 至 $IOV_{DD}$	-0.3 V 至 +0.3 V
AGND 至 DGND	-0.3 V 至 +0.3 V
$IOV_{DD}$ 至 IOGND， $AV_{DD}$ 至 AGND	-0.3 V 至 +6 V
数字输入电压至 IOGND	-0.3 V 至 +5.3 V
数字输出电压至 IOGND	-0.3 V 至 $IOV_{DD} + 0.3$ V
$V_{REF-2.5}$ 和 $V_{REF-1.2}$ 至 AGND	-0.3 V 至 $AV_{DD} + 0.3$ V
模拟输入至 AGND	-0.3 V 至 $AV_{DD} + 0.3$ V
模拟输出至 AGND	-0.3 V 至 $AV_{DD} + 0.3$ V
工业温度范围	-10°C 至 +95°C
存储温度范围	-65°C 至 +150°C
结温	150°C
$\theta_{JA}$ 热阻	
108引脚CSP_BGA	40°C/W
回流焊峰值温度	
锡铅体系(10 s至30 s)	240°C
RoHS体系(20 s至40 s)	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。这只是额定最值，并不能以这些条件或者在任何其他超出本技术规范操作章节中所示规格的条件下，推断器件能否正常工作。长期在绝对最大额定值条件下工作会影响器件的可靠性。

任何时候只能使用一个绝对最大额定值。

### 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。

## 引脚配置和功能描述

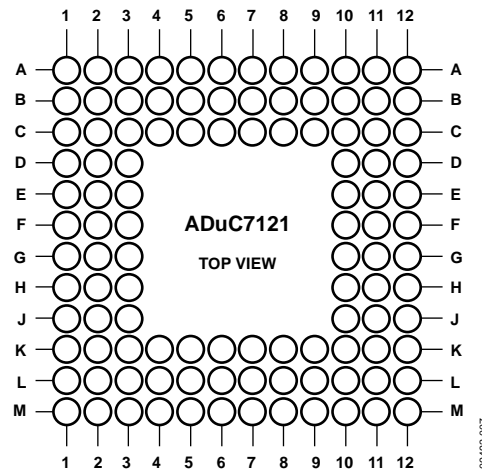


图7. 引脚配置

表9. 引脚功能描述

引脚编号	引脚名称	类型 <sup>1</sup>	说明
C12	RST	I	复位输入(低电平有效)。
D11	P0.0/SCL0/PLAI[5]	I/O	通用输入和输出端口0.0 (P0.0)。 I <sup>2</sup> C的I <sup>2</sup> C接口串行时钟(SCL0)。 输入单元5的可编程逻辑阵列(PLAI[5])。
E11	P0.1/SDA0/PLAI[4]	I/O	通用输入和输出端口0.1 (P0.1)。 I <sup>2</sup> C的I <sup>2</sup> C接口串行数据(SDA0)。 输入单元4的可编程逻辑阵列(PLAI[4])。
C3	P0.2/SPICLK/ADC <sub>BUSY</sub> /PLAO[13]	I/O	通用输入和输出端口0.2 (P0.2)。 SPI时钟(SPICLK)。 ADC状态(ADC <sub>BUSY</sub> )。 输出单元13的可编程逻辑阵列(PLAO[13])。
D3	P0.3/MISO/PLAO[12]/SYNC	I/O	通用输入和输出端口0.3 (P0.3)。 SPI主机输入、从机输出(MISO)。 输出单元12的可编程逻辑阵列(PLAO[12])。 同步复位(SYNC)。利用外部源同步复位PWM计数器的输入。
E3	P0.4/MOSI/PLAI[11]/TRIP	I/O	通用输入和输出端口0.4 (P0.4)。 SPI主机输出、从机输入(MOSI)。 输入单元11的可编程逻辑阵列(PLAI[11])。 PWM触发中断(TRIP)。引脚E3的TRIP功能是允许触发PWM中断的输入。
F3	P0.5/ $\overline{CS}$ /PLAI[10]/ADC <sub>CONVST</sub>	I/O	通用输入和输出端口0.5 (P0.5)。 SPI从机选择输入( $\overline{CS}$ )。 输入单元10的可编程逻辑阵列(PLAI[10])。 ADC转换(ADC <sub>CONVST</sub> )。引脚F3的ADC <sub>CONVST</sub> 功能利用PLA或定时器输出启动ADC转换。
G3	P0.6/ $\overline{MRST}$ /PLAI[2]	I/O	通用输入和输出端口0.6 (P0.6)。 上电复位输出( $\overline{MRST}$ )。 输入单元2的可编程逻辑阵列(PLAI[2])。
G10	P0.7/ $\overline{TRST}$ /PLAI[3]	I/O	通用输入和输出端口0.7 (P0.7)。 JTAG测试端口输入, 测试复位( $\overline{TRST}$ )。调试和下载访问。 输入单元3的可编程逻辑阵列(PLAI[3])。



引脚编号	引脚名称	类型 <sup>1</sup>	说明
C2	P1.0/SIN/SCL1/PLAI[7]	I/O	通用输入和输出端口1.0 (P1.0)。串行输入，接收数据，UART (SIN)。I <sup>2</sup> C1的I <sup>2</sup> C接口串行时钟(SCL1)。输入单元7的可编程逻辑阵列(PLAI[7])。
D2	P1.1/SOUT/SDA1/PLAI[6]	I/O	通用输入和输出端口1.1 (P1.1)。串行输出，发送数据，UART (SOUT)。I <sup>2</sup> C1的I <sup>2</sup> C接口串行数据(SDA1)。输入单元6的可编程逻辑阵列(PLAI[6])。
C10	P1.2/TDI/PLAO[15]	DI	通用输入和输出端口1.2 (P1.2)。JTAG测试端口输入，测试数据输入(TDI)。引脚C10的TDI功能用于调试和下载访问。输出单元15的可编程逻辑阵列(PLAO[15])。
D10	P1.3/TDO/PLAO[14]	DO	通用输入和输出端口1.3 (P1.3)。JTAG测试端口输出，测试数据输出(TDO)。引脚D10的TDO功能用于调试和下载访问。输出单元14的可编程逻辑阵列(PLAO[14])。
H3	P1.4/PWM1/ECLK/XCLK/PLAI[8]	I/O	通用输入和输出端口1.4 (P1.4)。脉宽调制器1输出(PWM1)。基本系统时钟输出(ECLK)。基本系统时钟输入(XCLK)。输入单元8的可编程逻辑阵列(PLAI[8])。
J3	P1.5/PWM2/PLAI[9]	I/O	通用输入和输出端口1.5 (P1.5)。脉宽调制器2输出(PWM2)。输入单元9的可编程逻辑阵列(PLAI[9])。
B3	P1.6/PLAO[5]	I/O	通用输入和输出端口1.6 (P1.6)。输出单元5的可编程逻辑阵列(PLAO[5])。
B2	P1.7/PLAO[4]	I/O	通用输入和输出端口1.7 (P1.7)。输出单元4的可编程逻辑阵列(PLAO[4])。
F11	P2.0/IRQ0/PLAI[13]	I/O	通用输入输出端口2.0 (P2.0)/外部中断请求0，高电平有效。输入单元13的可编程逻辑阵列(PLAI[13])。
G11	P2.1/IRQ1/PLAI[12]	I/O	通用输入和输出端口2.1 (P2.1)。外部中断请求1，高电平有效(IRQ1)。输入单元12的可编程逻辑阵列(PLAI[12])。
H11	P2.2/PLAI[1]	I/O	通用输入和输出端口2.2 (P2.2)。输入单元1的可编程逻辑阵列(PLAI[1])。
J11	P2.3/IRQ2/PLAI[14]	I/O	通用输入和输出端口2.3 (P2.3)。外部中断请求2，高电平有效(IRQ2)。输入单元14的可编程逻辑阵列(PLAI[14])。
H10	P2.4/PWM5/PLAO[7]	I/O	通用输入和输出端口2.4 (P2.4)。脉宽调制器5输出(PWM5)。输出单元7的可编程逻辑阵列(PLAO[7])。
J10	P2.5/PWM6/PLAO[6]	I/O	通用输入和输出端口2.5 (P2.5)。脉宽调制器6输出(PWM6)。输出单元6的可编程逻辑阵列(PLAO[6])。
C1	P2.6/IRQ3/PLAI[15]	I/O	通用输入和输出端口2.6 (P2.6)。外部中断请求3，高电平有效(IRQ3)。输入单元15的可编程逻辑阵列(PLAI[15])。
C9	P2.7/PLAI[0]	I/O	通用输入和输出端口2.7 (P2.7)。
C4	P3.0/PLAO[0]	I/O	通用输入和输出端口3.0 (P3.0)。输出单元0的可编程逻辑阵列(PLAO[0])。
C11	P3.1/PLAO[1]	I/O	通用输入和输出端口3.1 (P3.1)。输出单元1的可编程逻辑阵列(PLAO[1])。

# ADuC7121

引脚编号	引脚名称	类型 <sup>1</sup>	说明
D1	P3.2/IRQ4/PWM3/PLAO[2]	I/O	通用输入和输出端口3.2 (P3.2)。外部中断请求4，高电平有效(IRQ4)。脉宽调制器3输出(PWM3)。输出单元2的可编程逻辑阵列(PLAO[2])。
E1	P3.3/IRQ5/PWM4/PLAO[3]	I/O	通用输入和输出端口3.3 (P3.3)。外部中断请求5，高电平有效(IRQ5)。脉宽调制器4输出(PWM4)。输出单元3的可编程逻辑阵列(PLAO[3])。
E2	P3.4/PLAO[8]	I/O	通用输入和输出端口3.4 (P3.4)。输出单元8的可编程逻辑阵列(PLAO[8])。
F2	P3.5/PLAO[9]	I/O	通用输入和输出端口3.5 (P3.5)。输出单元9的可编程逻辑阵列(PLAO[9])。
D12	P3.6/PLAO[10]	I/O	通用输入和输出端口3.6 (P3.6)。输出单元10的可编程逻辑阵列(PLAO[10])。
E12	P3.7/ $\overline{\text{BM}}$ /PLAO[11]	I/O	通用输入和输出端口3.7 (P3.7)。输出单元11的可编程逻辑阵列(PLAO[11])。
L8	V <sub>REF_2.5</sub>	AI/O	2.5 V基准电压输出和外部2.5 V基准电压输入。
L5	V <sub>REF_1.2</sub>	AI/O	1.2 V基准电压输出和外部1.2 V基准电压输入。不能用于向外部流出电流。
B8	I <sub>REF</sub>	AI/O	产生IDAC的基准电流。由外部电阻R <sub>EXT</sub> 设置。
K6	BUF_VREF1	AO	缓冲2.5 V。BUF_VREF1的最大负载是1.2 mA。
K7	BUF_VREF2	AO	缓冲2.5 V。BUF_VREF2的最大负载是1.2 mA。
L6	PADC0P	AI	PGA通道0+。
M5	PADC0N	AI	PGA通道0-。
L7	PADC1P	AI	PGA通道1+。
M8	PADC1N	AI	PGA通道1-。
K5	NC	NC	不连接。请勿连接该引脚。
K4	NC	NC	不连接。请勿连接该引脚。
M4	NC	NC	不连接。请勿连接该引脚。
L4	NC	NC	不连接。请勿连接该引脚。
K3	ADC4	AI	单端或差分模拟输入4。
M3	ADC5	AI	单端或差分模拟输入5。
M10	ADC6	AI	单端或差分模拟输入6。
M9	ADC7	AI	单端或差分模拟输入7。
L9	ADC8	AI	单端或差分模拟输入8。
K9	ADC9	AI	单端或差分模拟输入9。
K8	ADC10/AINCM	AI	单端或差分模拟输入10 (ADC10)。共模(AINCM)。此引脚的共模功能用于伪差分输入。
K1	DAC0	AO	12位DAC0输出。
K2	DAC1	AO	12位DAC1输出。
J2	NC	NC	不连接。请勿连接该引脚。
L2	NC	NC	不连接。请勿连接该引脚。
M2	NC	NC	不连接。请勿连接该引脚。
L3	NC	NC	不连接。请勿连接该引脚。
M11	DAC2	AO	12位DAC2输出。
L11	NC	NC	不连接。请勿连接该引脚。
L10	NC	NC	不连接。请勿连接该引脚。
K10	NC	NC	不连接。请勿连接该引脚。
K11	NC	NC	不连接。请勿连接该引脚。
K12	DAC3	AO	12位DAC3输出。
B5	IDAC4	AO	IDAC4输出。此引脚的输出为20 mA。
C6	PVDD_IDAC4	S	IDAC4的2.0 V电源。
A6	C <sub>DAMP_IDAC4</sub>	AI	IDAC4的阻尼电容引脚。

引脚编号	引脚名称	类型 <sup>1</sup>	说明
A8	IDAC2	AO	IDAC2输出。此引脚的输出为45 mA。
A7	PVDD_IDAC2	S	IDAC2的2.0 V电源。
C8	C <sub>DAMP_IDAC2</sub>	AI	IDAC2的阻尼电容引脚。
A5	IDAC3	AO	IDAC3输出。此引脚的输出为80 mA。
C5	PVDD_IDAC3	S	IDAC3的2.0 V电源。
B4	C <sub>DAMP_IDAC3</sub>	AI	IDAC3的阻尼电容。
A4	IDAC1	AO	IDAC1输出。此引脚的输出为200 mA。
A1	IDAC1	AO	IDAC1输出。此引脚的输出为200 mA。
A3	PVDD_IDAC1	S	IDAC1的电源。
A2	PVDD_IDAC1	S	IDAC1的电源。
B1	C <sub>DAMP_IDAC1</sub>	AI	IDAC1的阻尼电容。
A12	IDAC0	AO	IDAC0输出。此引脚的输出为250 mA。
A9	IDAC0	AO	IDAC0输出。此引脚的输出为250 mA。
A11	PVDD_IDAC0	S	IDAC0的电源。
A10	PVDD_IDAC0	S	IDAC0的电源。
B12	C <sub>DAMP_IDAC0</sub>	AI	IDAC0的阻尼电容引脚。
B11	IDAC_TST	AI/O	IDAC测试。
B10	PGND	S	电源地。
B9	PGND	S	电源地。
M1	AGND	S	模拟地。
M6	AGND	S	模拟地。
L1	AVDD	S	模拟电源(3.3 V)。
M7	AVDD	S	模拟电源(3.3 V)。
M12	AGND	S	模拟地。
B6	AGND	S	模拟地。
L12	AVDD	S	模拟电源(3.3 V)。
C7	AVDD	S	模拟电源(3.3 V)。
B7	AVDD_IDAC	S	内部IDAC的2.5 V LDO稳压器输出。必须在此引脚与AGND之间连接一个470 nF电容。
G1	DVDD	S	2.6 V片内LDO稳压器的输出。必须在此引脚与DGND之间连接一个470 nF电容。
G12	DVDD	S	2.6 V片内LDO稳压器的输出。必须在此引脚与DGND之间连接一个470 nF电容。
F1	DGND	S	数字地。
F12	DGND	S	数字地。
H1	IOVDD	S	3.3 V GPIO电源。
J1	IOGND	S	3.3 V GPIO地。
H12	IOVDD	S	3.3 V GPIO电源。
J12	IOGND	S	3.3 V GPIO地。
G2	XTALO	DO	晶振反相器输出。如果不使用外部晶体，此引脚可以保持断开状态。
H2	XTALI	DI	晶振反相器输入和内部时钟发生器电路输入。如果不使用外部晶体，应将此引脚接到DGND系统地。
F10	TCK	DI	JTAG测试端口输入，测试时钟。调试和下载访问。
E10	TMS	DI	JTAG测试端口输入，测试模式选择。调试和下载访问。

<sup>1</sup> A表示模拟，D表示数字，I表示输入，O表示输出，S表示电源，NC表示不连接。

## 术语

### ADC规格

#### 积分非线性

积分非线性(INL)是指转换结果编码与通过ADC传递函数端点的直线的最大偏差。传递函数端点是指，在零电平位置比第一个编码的跃变点低 $\frac{1}{2}$  LSB的点，以及在满量程位置比最后一个编码的跃变点高 $\frac{1}{2}$  LSB的点。

#### 微分非线性

微分非线性(DNL)是指ADC中任意两个相邻码之间所测得变化值与理想的1 LSB变化值之间的差异。

#### 失调误差

失调误差是指第一个转换编码(从0000...000到0000...001)的跃变点与理想点 $+\frac{1}{2}$  LSB之间的偏差。

#### 增益误差

增益误差是指在失调误差调零之后，最后一个转换编码的跃变点与理想AIN电压(满量程 - 1.5 LSB)的偏差。

#### 信纳比

信纳比(SINAD)是指在ADC输出端测得的信号对噪声及失真比。这里的信号是基波幅值的均方根值。噪声为除了直流信号以外一直到半采样频率( $f_s/2$ )的所有非基波信号均方根和。

在数字化过程中，这个比值的大小取决于量化级数：量化级数越多，量化噪声就越小。

对于一个正弦波输入的理想N位转换器，信纳比理论值计算公式为：

$$\text{信号与(噪声 + 失真)比值} = (6.02N + 1.76) \text{ dB}$$

因此，对于12位转换器，该值为74 dB。

#### 总谐波失真

总谐波失真(THD)是指所有谐波均方根和与基波的比值。

#### DAC技术规格

##### 相对精度

也被称作端点线性度，相对精度是指DAC输出与通过DAC端点的传递函数直线之间的最大偏差。在零点误差和满量程误差调零后才可进行相对精度测量。

##### 输出电压建立时间

它是指对于一个满量程输入变化，DAC输出稳定在1 LSB变化范围内所需时间。

## ARM7TDMI内核概览

ARM7™内核为32位精简指令集计算机(RISC)。指令和数据使用单32位总线。数据的长度可以是8位、16位或32位。指令字的长度为32位。

ARM7TDMI®采用ARM7内核，具有如下4种附加特性：

- 支持16位的thumb指令集(T)
- 支持调试(D)
- 支持长乘(M)
- 包含一个支持嵌入式系统调试的EmbeddedICE™模块(I)

### THUMB模式(T)

一条ARM®指令的长度为32位。ARM7TDMI处理器支持压缩至16位的第二指令集，即Thumb®指令集。用Thumb指令集替代ARM指令集，可以更为快速地从16位存储器执行代码并且实现更高的代码密度。这就使得ARM7TDMI内核尤其适用于嵌入式系统。

然而，Thumb模式有以下两个缺点：

- 对于同一工作，Thumb代码通常需要更多指令。因此，如果更强调时效性，ARM代码更适合用来优化代码性能。
- Thumb指令集并不包含异常处理的所有指令，所以如果异常发生在Thumb状态，处理器会自动切换到ARM代码。

关于内核架构、编程模块、ARM和ARM Thumb指令集的具体内容，请参阅ARM7TDMI用户手册。

### 长乘(M)

ARM7TDMI指令集包括四个额外的指令，分别为得到64位结果的32位与32位相乘指令；得到64位结果的32位与32位乘加(MAC)指令。得到这些结果比标准的ARM7内核所需的时钟周期更少。

### EmbeddedICE (I)

EmbeddedICE支持内核片内调试。EmbeddedICE模块包含断点和观察点寄存器，在调试时这些寄存器可使代码中止执行。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，处理器中断，并进入调试状态。进入调试状态时，就可以检查处理器寄存器、Flash/EE、SRAM和存储器映射寄存器的状态。

### 异常

ARM支持5种类型的异常，并且每一种异常模式有一种优先处理器模式。这5种异常为：

- 正常中断或IRQ。这种异常用于内部和外部事件的通用中断处理。
- 快速中断或FIQ。这是用于数据传输或低延迟时间通道处理。FIQ的优先级高于IRQ。
- 存储器中止。
- 尝试执行未定义指令。
- 软件中断指令(SWI)。它通常用于通知操作系统。

典型情况下，程序员定义中断为IRQ，但是为了得到更高优先级的中断，即得到更快响应时间，程序员可以定义中断为FIQ。

## ARM寄存器

ARM7TDMI共有37个寄存器：31个通用寄存器和6个状态寄存器。每一个工作模式都有专门的寄存器组。

当编写用户级程序时，15个通用32位寄存器(R0-R14)、程序计数器(R15)和当前程序状态寄存器(CPSR)是可用的。余下的寄存器只用于系统级编程和异常处理。

异常发生后，异常模式专用的寄存器将取代某些标准寄存器。所有的异常模式有各自的替换寄存器组，用于堆栈指针(R13)和链接寄存器(R14)，如图8所示。快速中断模式有更多的寄存器(R8到R12)用于快速中断处理。这意味着无需先保存或者重新保存这些寄存器，就可以进行中断处理，因此在中断处理中可以节省至关重要的时间。

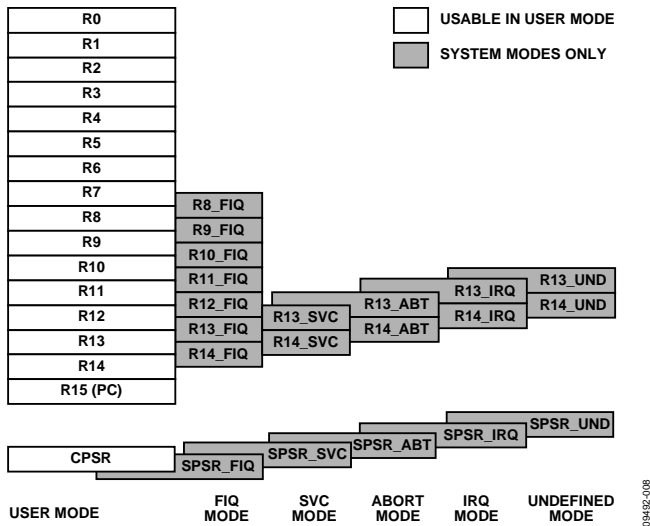


图 8. 寄存器结构图

更多关于编程模式和ARM7TDMI内核架构的信息可通过以下ARM公司的文件获得：

- ARM DDI 0029G, ARM7TDMI技术参考手册
- ARM DDI 0100, ARM架构参考手册

## 中断延迟时间

快速中断请求(FIQ)的最大延迟时间包含：

- 请求通过同步器的最长时间。
- 最长指令完成所需的时间；最长指令是LDM，用于加载所有寄存器，包括PC。
- 数据中止入口时间。
- FIQ入口时间。

在这个时间段的末尾，ARM7TDMI执行0X1C(FIQ中断向量地址)处的指令。最长总延迟时间为50个处理器周期，在系统采用连续41.78 MHz处理器时钟时，略小于1.2 μs。

中断请求(IRQ)最大延迟时间计算方法也类似，但必须考虑到FIQ优先级更高，可能任意延长进入IRQ处理例行程序的时间。如果不使用LDM命令，这个时间可以缩短到42个周期。一些编译器可以选择不使用这个命令进行编译。另一个选择是在Thumb模式下运行器件，可以将时间缩短至22个周期。

用于FIQ或IRQ的最小中断延迟时间总共有5个周期，包括请求通过同步器的最短时间和进入异常模式的时间。

注意优先模式中(例如执行中断服务程序)，ARM7TDMI通常运行于32位的ARM模式。

## 存储器结构

ADuC7121有三个独立存储器模块：8 kB的SRAM和两个64 kB的片内Flash/EE存储器。片内Flash/EE存储器有126 kB可供用户使用，剩余2 kB保留用于厂家配置的引导页面。这两个存储器模块的映射如图9所示。

注意默认情况下，复位之后，Flash/EE存储器被镜像到地址0x00000000。将REMAP寄存器的位0清0，可以重新把SRAM映射到0x00000000。这种重映射功能在Flash/EE存储器部分有更详细描述。

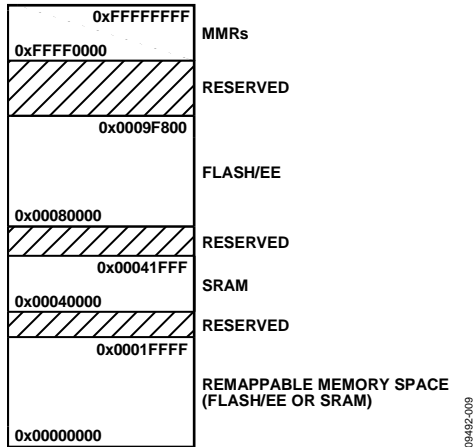


图9. 物理存储器映射图

### 存储器访问

ARM7内核把存储器看成是232个字节的一个线性阵列。不同的存储器模块映射如图9所示。

ADuC7121存储器被配置成从小到大顺序格式：LSB位于最低字节地址，MSB位于最高字节地址。

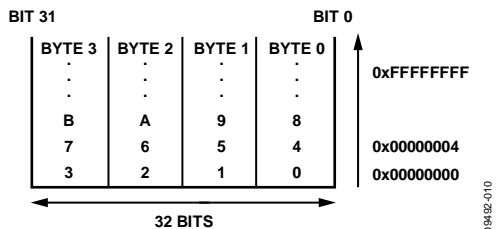


图10. 从小到大顺序格式

### FLASH/EE存储器

128 kB的Flash/EE分为两个32 k x 16位的模块。模块0起始地址为0x90000，结束地址为0x9F700。在该模块中，31 k x 16位是用户空间，1 k x 16位保留用于工厂配置的引导页面。Flash/EE存储器的页面大小为512个字节。

模块1起始地址为0x80000，结束地址为0x90000。该64 kB模块配置为32 k x 16位，整个模块均作用用户空间。

126 kB的Flash/EE存储器可以存储用户代码和非易失性数据。数据和代码之间没有区别，因为ARM代码及数据共用同一空间。Flash/EE存储器的实际宽度为16位，这意味着在ARM模式(32位指令)下每次取指令必须两次访问Flash/EE存储器。因此，当从Flash/EE存储器中执行程序时，建议使用Thumb模式来优化存取速度。以Thumb模式存取Flash/EE存储器的最大速度为41.78 MHz，而相应的以全ARM模式为20.89 MHz(参见“SRAM和FLASH/EE执行时间”部分)。

### SRAM

用户可以使用8 kB的SRAM，它的组织形式为2k x 32位，即2k字。如果SRAM被配置成32位宽的存储器阵列，ARM代码可以直接在SRAM中以41.78MHz的速度执行(参见“SRAM和FLASH/EE执行时间”部分)。

### 存储器映射寄存器

存储器映射寄存器(MMR)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。

MMR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11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10至表27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：高级性能总线(AHB)用于系统模块，高级外设总线(APB)用于低性能外围设备。访问AHB需要一个周期，访问APB需要两个周期。除了Flash/EE存储器和通用输入输出端口以外，ADuC7121中的所有外围设备均位于APB上。

# ADuC7121

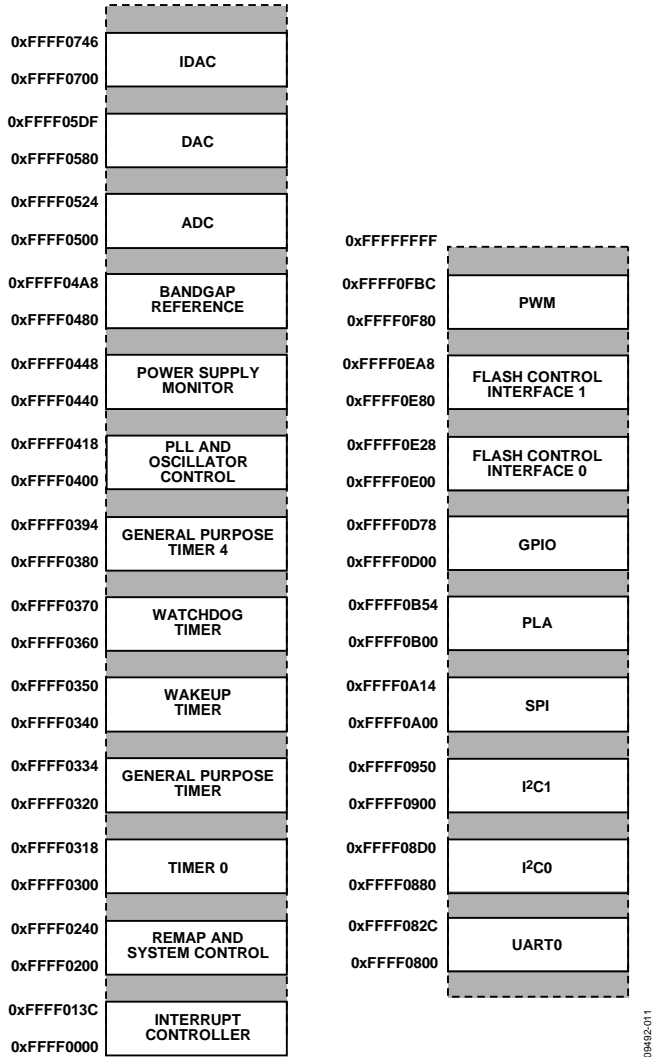


图 11. 存储器映射寄存器

## 完整MMR列表

注意：“访问类型”栏对应于读取或写入寄存器的访问时间，其中R表示读取，W表示写入，R/W表示读/写。它取决于用来访问外设的AMBA总线。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。

表10. IRQ基地址 = 0xFFFF0000

地址	名称	字节	访问类型	周期
0x0000	IRQSTA	4	R	1
0x0004	IRQSIG	4	R	1
0x0008	IRQEN	4	R/W	1
0x000C	IRQCLR	4	R/W	1
0x0010	SWICFG	4	W	1
0x0014	IRQBASE	4	R/W	1
0x001C	IRQVEC	4	R/W	1
0x0020	IRQP0	4	R/W	1
0x0024	IRQP1	4	R/W	1
0x0028	IRQP2	4	R/W	1

地址	名称	字节	访问类型	周期
0x002C	IRQP3	4	R/W	1
0x0030	IRQCONN	1	R/W	1
0x0034	IRQCONE	4	R/W	1
0x0038	IRQCLRE	1	W	1
0x003C	IRQSTAN	1	R/W	1
0x0100	FIQSTA	4	R	1
0x0104	FIQSIG	4	R	1
0x0108	FIQEN	4	R/W	1
0x010C	FIQCLR	4	W	1
0x011C	FIQVEC	4	R	1
0x013C	FIQSTAN	1	R/W	1

表11. 系统控制基地址 = 0xFFFF0200

地址	名称	字节	访问类型	周期
0x0220	REMAP	1	R/W	1
0x0230	RSTSTA	1	R	1
0x0234	RSTCLR	1	W	1
0x0248	RSTCFGKEY0	1	W	1
0x024C	RSTCFG	1	R/W	1
0x0250	RSTCFGKEY1	1	W	1

表12. 定时器基地址 = 0xFFFF0300

地址	名称	字节	访问类型	周期
0x0300	T0LD	2	R/W	2
0x0304	T0VAL0	2	R	2
0x0308	T0VAL1	4	R	2
0x030C	T0CON	4	R/W	2
0x0310	T0CLR1	1	W	2
0x0314	T0CAP	2	R	2
0x0320	T1LD	4	R/W	2
0x0324	T1VAL	4	R	2
0x0328	T1CON	4	R/W	2
0x032C	T1CLR1	1	W	2
0x0330	T1CAP	4	R	2
0x0340	T2LD	4	R/W	2
0x0344	T2VAL	4	R	2
0x0348	T2CON	4	R/W	2
0x034C	T2CLR1	1	W	2
0x0360	T3LD	2	R/W	2
0x0364	T3VAL	2	R	2
0x0368	T3CON	2	R/W	2
0x036C	T3CLR1	1	W	2
0x0380	T4LD	4	R/W	2
0x0384	T4VAL	4	R	2
0x0388	T4CON	4	R/W	2
0x038C	T4CLR1	1	W	2
0x0390	T4CAP	4	R	2

表13. PLL基地址 = 0xFFFF0400

地址	名称	字节	访问类型	周期
0x0404	POWKEY1	2	W	2
0x0408	POWCON	1	R/W	2
0x040C	POWKEY2	2	W	2
0x0410	PLLKEY1	2	W	2
0x0414	PLLCON	1	R/W	2
0x0418	PLLKEY2	2	W	2



表14. PSM基地址 = 0xFFFF0440

地址	名称	字节	访问类型	周期
0x0440	PSMCON	2	R/W	2

表15. 基准电压基地址 = 0xFFFF0480

地址	名称	字节	访问类型	周期
0x0480	REFCON	1	R/W	2

表16. ADC基地址 = 0xFFFF0500

地址	名称	字节	访问类型	周期
0x0500	ADCCON	4	R/W	2
0x0504	ADCCP	1	R/W	2
0x0508	ADCCN	1	R/W	2
0x050C	ADCSTA	1	R	2
0x0510	ADCDAT	4	R	2
0x0514	ADCRST	1	W	2
0x0518	ADCGN	2	R/W	2
0x051C	ADCOF	2	R/W	2
0x0520	PGA_GN	2	R/W	2

表17. DAC基地址 = 0xFFFF0580

地址	名称	字节	访问类型	周期
0x0580	DAC0CON	2	R/W	2
0x0584	DAC0DAT	4	R/W	2
0x0588	DAC1CON	2	R/W	2
0x058C	DAC1DAT	4	R/W	2
0x05B0	DAC2CON	2	R/W	2
0x05B4	DAC2DAT	4	R/W	2
0x05D8	DAC3CON	2	R/W	2
0x05DC	DAC3DAT	4	R/W	2

表18. IDAC基地址 = 0xFFFF0700

地址	名称	字节	访问类型	周期
0x0700	IDAC0CON	2	R/W	2
0x0704	IDAC0DAT	4	R/W	2
0x0708	IDAC0BW	1	R/W	2
0x070C	IDAC1CON	2	R/W	2
0x0710	IDAC1DAT	4	R/W	2
0x0714	IDAC1BW	1	R/W	2
0x0718	IDAC2CON	2	R/W	2
0x071C	IDAC2DAT	4	R/W	2
0x0720	IDAC2BW	1	R/W	2
0x0724	IDAC3CON	2	R/W	2
0x0728	IDAC3DAT	4	R/W	2
0x072C	IDAC3BW	1	R/W	2
0x0730	IDAC4CON	2	R/W	2
0x0734	IDAC4DAT	4	R/W	2
0x0738	IDAC4BW	1	R/W	2
0x073C	TSDCON	1	R/W	2
0x0740	IDACSTA	1	R/W	2
0x0744	IDAC0PULLDOWN	1	R/W	2

表19. UART0基地址 = 0xFFFF0800

地址	名称	字节	访问类型	周期
0x0800	COMTX	1	W	2
	COMRX	1	R	2
	COMDIV0	1	R/W	2
0x0804	COMIEN0	1	R/W	2
	COMDIV1	1	R/W	2
0x0808	COMIID0	1	R	2
0x080C	COMCON0	1	R/W	2
0x0810	COMCON1	1	R/W	2
0x0814	COMSTA0	1	R	2
0x082C	COMDIV2	2	R/W	2

表20. I<sup>2</sup>C0基地址 = 0xFFFF0880

地址	名称	字节	访问类型	周期
0x0880	I2C0MCTL	2	R/W	2
0x0884	I2C0MSTA	2	R	2
0x0888	I2C0MRX	1	R	2
0x088C	I2C0MTX	2	W	2
0x0890	I2C0MCNT0	2	R/W	2
0x0894	I2C0MCNT1	1	R	2
0x0898	I2C0ADRO	1	R/W	2
0x089C	I2C0ADR1	1	R/W	2
0x08A0	I2C0SBYTE	1	R/W	2
0x08A4	I2C0DIV	2	R/W	2
0x08A8	I2C0SCTL	2	R/W	2
0x08AC	I2C0SSTA	2	R	2
0x08B0	I2C0SRX	1	R	2
0x08B4	I2C0STX	1	W	2
0x08B8	I2C0ALT	1	R/W	2
0x08BC	I2C0ID0	1	R/W	2
0x08C0	I2C0ID1	1	R/W	2
0x08C4	I2C0ID2	1	R/W	2
0x08C8	I2C0ID3	1	R/W	2
0x08CC	I2C0FSTA	1	R/W	2

表21. I<sup>2</sup>C1基地址 = 0xFFFF0900

地址	名称	字节	访问类型	周期
0x0900	I2C1MCTL	2	R/W	2
0x0904	I2C1MSTA	2	R	2
0x0908	I2C1MRX	1	R	2
0x090C	I2C1MTX	2	W	2
0x0910	I2C1MCNT0	2	R/W	2
0x0914	I2C1MCNT1	1	R	2
0x0918	I2C1ADRO	1	R/W	2
0x091C	I2C1ADR1	1	R/W	2
0x0920	I2C1SBYTE	1	R/W	2
0x0924	I2C1DIV	2	R/W	2
0x0928	I2C1SCTL	2	R/W	2
0x092C	I2C1SSTA	2	R	2
0x0930	I2C1SRX	1	R	2
0x0934	I2C1STX	1	W	2
0x0938	I2C1ALT	1	R/W	2
0x093C	I2C1ID0	1	R/W	2

# ADuC7121

地址	名称	字节	访问类型	周期
0x0940	I2C1ID1	1	R/W	2
0x0944	I2C1ID2	1	R/W	2
0x0948	I2C1ID3	1	R/W	2
0x094C	I2C1FSTA	1	R/W	2

**表22. SPI基地址 = 0xFFFF0A00**

地址	名称	字节	访问类型	周期
0x0A00	SPISTA	2	R	2
0x0A04	SPIRX	1	R	2
0x0A08	SPITX	1	W	2
0x0A0C	SPIDIV	1	R/W	2
0x0A10	SPICON	2	R/W	2

**表23. PLA基地址 = 0xFFFF0B00**

地址	名称	字节	访问类型	周期
0x0B00	PLAELM0	2	R/W	2
0x0B04	PLAELM1	2	R/W	2
0x0B08	PLAELM2	2	R/W	2
0x0B0C	PLAELM3	2	R/W	2
0x0B10	PLAELM4	2	R/W	2
0x0B14	PLAELM5	2	R/W	2
0x0B18	PLAELM6	2	R/W	2
0x0B1C	PLAELM7	2	R/W	2
0x0B20	PLAELM8	2	R/W	2
0x0B24	PLAELM9	2	R/W	2
0x0B28	PLAELM10	2	R/W	2
0x0B2C	PLAELM11	2	R/W	2
0x0B30	PLAELM12	2	R/W	2
0x0B34	PLAELM13	2	R/W	2
0x0B38	PLAELM14	2	R/W	2
0x0B3C	PLAELM15	2	R/W	2
0x0B40	PLACLK	1	R/W	2
0x0B44	PLAIRQ	2	R/W	2
0x0B48	PLAADC	4	R/W	2
0x0B4C	PLADIN	4	R/W	2
0x0B50	PLAOUT	4	R	2
0x0B54	PLALCK	1	W	2

**表24. GPIO基地址 = 0xFFFF0D00**

地址	名称	字节	访问类型	周期
0x0D00	GPOCON	4	R/W	1
0x0D04	GP1CON	4	R/W	1
0x0D08	GP2CON	4	R/W	1
0x0D0C	GP3CON	4	R/W	1
0x0D20	GPODAT	4	R/W	1
0x0D24	GPOSET	1	W	1
0x0D28	GPOCLR	1	W	1
0x0D2C	GP0PAR	4	R/W	1
0x0D30	GP1DAT	4	R/W	1
0x0D34	GP1SET	1	W	1
0x0D38	GP1CLR	1	W	1
0x0D3C	GP1PAR	4	R/W	1
0x0D40	GP2DAT	4	R/W	1

地址	名称	字节	访问类型	周期
0x0D44	GP2SET	1	W	1
0x0D48	GP2CLR	1	W	1
0x0D4C	GP2PAR	4	R/W	1
0x0D50	GP3DAT	4	R/W	1
0x0D54	GP3SET	1	W	1
0x0D58	GP3CLR	1	W	1
0x0D5C	GP3PAR	4	R/W	1

**表25. Flash/EE模块0基地址 = 0xFFFF0E00**

地址	名称	字节	访问类型	周期
0x0E00	FEE0STA	1	R	1
0x0E04	FEE0MOD	1	R/W	1
0x0E08	FEE0CON	1	R/W	1
0x0E0C	FEE0DAT	2	R/W	1
0x0E10	FEE0ADR	2	R/W	1
0x0E18	FEE0SGN	3	R	1
0x0E1C	FEE0PRO	4	R/W	1
0x0E20	FEE0HID	4	R/W	1

**表26. Flash/EE模块1基地址 = 0xFFFF0E80**

地址	名称	字节	访问类型	周期
0x0E80	FEE1STA	1	R	1
0x0E84	FEE1MOD	1	R/W	1
0x0E88	FEE1CON	1	R/W	1
0x0E8C	FEE1DAT	2	R/W	1
0x0E90	FEE1ADR	2	R/W	1
0x0E98	FEE1SGN	3	R	1
0x0E9C	FEE1PRO	4	R/W	1
0x0EA0	FEE1HID	4	R/W	1

**表27. PWM基地址 = 0xFFFF0F80**

地址	名称	字节	访问类型	周期
0x0F80	PWMCON1	2	R/W	2
0x0F84	PWM1COM1	2	R/W	2
0x0F88	PWM1COM2	2	R/W	2
0x0F8C	PWM1COM3	2	R/W	2
0x0F90	PWM1LEN	2	R/W	2
0x0F94	PWM2COM1	2	R/W	2
0x0F98	PWM2COM2	2	R/W	2
0x0F9C	PWM2COM3	2	R/W	2
0x0FA0	PWM2LEN	2	R/W	2
0x0FA4	PWM3COM1	2	R/W	2
0x0FA8	PWM3COM2	2	R/W	2
0x0FAC	PWM3COM3	2	R/W	2
0x0FB0	PWM3LEN	2	R/W	2
0x0FB4	PWMCON2	2	R/W	2
0x0FB8	PWMICLR	2	W	2

## ADC电路概览

该模数转换器(ADC)集成一个快速多通道12位ADC,工作电压为3.0 V至3.6 V,当时钟源为41.78 MHz时,其吞吐量最高可以达到1 MSPS。这个模块提供给用户一个多通道多路复用器、一个差分跟踪保持电路、一个片内基准电压源和一个ADC。

该ADC为一款基于两个电容DAC的12位逐次逼近型转换器。根据对输入信号的配置不同,ADC可以在如下三种模式下工作:

- 全差分模式,用于小信号和平衡信号。
- 单端模式,用于任意单端信号。
- 伪差分模式,用于任意单端信号,利用伪差分输入提供的共模抑制性能。

当工作在单端模式或伪差分模式下时,转换器模拟输入范围为0 V至 $V_{REF}$ 。在全差分模式下,输入信号共模电压 $V_{CM}$ 在0 V至 $AV_{DD}$ 之间,最大幅值不超过 $2V_{REF}$ (见图12)。

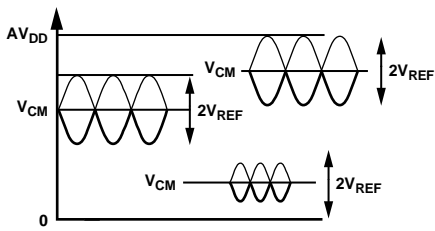


图12. 全差分模式下的平衡信号示例

片内提供一个高精度、低漂移、工厂校准的2.5 V基准电压源,也可以直接连接一个外部基准源,如“带隙基准电压源”部分所述。

单次或连续转换模式可由软件启动。外部 $ADC_{CONVST}$ 引脚、片内PLA生成的输出、定时器0或定时器1溢出也可以用来生成重复性的ADC转换触发信号。

ADC转换完成时,如果该信号尚未解除置位,则第二次转换会自动开始。另外,从片内带隙基准源输出且与绝对温度成正比的电压也可以通过ADC前端多路复用器路由,产生一个附加ADC通道输入。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,测量精度典型值为 $\pm 3^{\circ}C$ 。

ADuC7121经过修改,能够区分自身的ADC结构与ADuC702x系列中的其他器件。

PADC0x和PADC1x输入连接到PGA,提供从1到5的32步增益。其余通道可配置为单端或差分方式。ADC之前有一个缓冲器,用于测量内部通道。

## ADC传递函数

### 伪差分模式和单端模式

伪差分 and 单端模式下的输入电压范围均为0 V至 $V_{REF}$ ,并且输出编码均为标准二进制编码:

$$1 \text{ LSB} = FS/4096; \text{ 当 } V_{REF} = 2.5 \text{ V 时,} \\ 1 \text{ LSB} = 2.5 \text{ V}/4096 = 0.61 \text{ mV 或 } 610 \mu\text{V}.$$

理想编码转换发生在两个连续整数LSB值的中点,即 $1/2 \text{ LSB}$ 、 $3/2 \text{ LSB}$ 、 $5/2 \text{ LSB}$ 、...、 $FS - 3/2 \text{ LSB}$ 。理想输入/输出传递特性如图13所示。

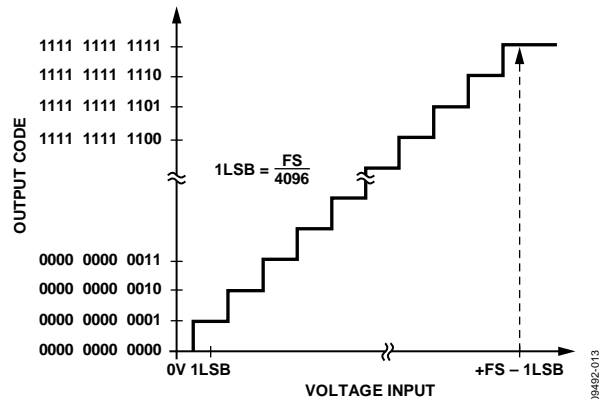


图13. 伪差分或单端模式下ADC传递函数

### 全差分模式

差分信号的幅值为输入 $V_{IN+}$ 和 $V_{IN-}$ 的信号差值(即 $V_{IN+} - V_{IN-}$ )。因此,差分信号的最大幅值为 $-V_{REF}$ 至 $+V_{REF}$ 峰峰值(即 $2 \times V_{REF}$ )。这与共模电压(CM)无关。共模电压是两个信号的平均值 $(V_{IN+} + V_{IN-})/2$ ,因此也是两个输入电压的中点,各输入的范围是 $CM \pm V_{REF}/2$ 。这一电压必须在外部设定,并且其范围随着 $V_{REF}$ 而变化(参见“驱动模拟输入”部分)。

在全差分模式下,输出编码为二进制补码:

$$1 \text{ LSB} = 2 V_{REF}/4096 \text{ 或 } 2 \times 2.5 \text{ V}/4096 = 1.22 \text{ mV 当 } \\ V_{REF} = 2.5 \text{ V}$$

输出结果为 $\pm 11$ 位,但要右移一位。这样,书写C语言代码时,就可以将ADCDAT中的结果声明为带符号整数。设计编码转换发生在两个连续整数LSB值的中点,即 $1/2 \text{ LSB}$ 、 $3/2 \text{ LSB}$ 、 $5/2 \text{ LSB}$ 、...、 $FS - 3/2 \text{ LSB}$ 。理想输入/输出传递特性如图14所示。

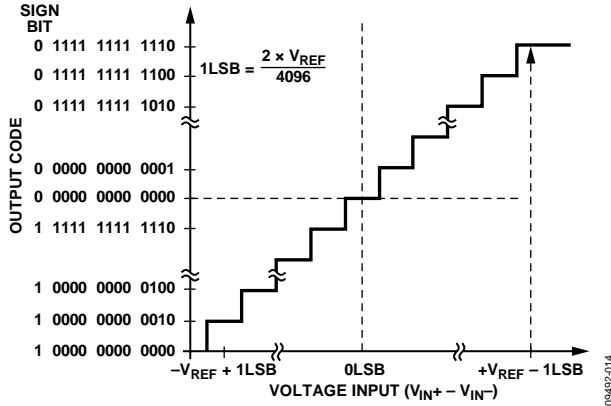


图14. 差分模式下ADC传递函数

## PADC0x/PADC1x引脚

PADC0x和PADC1x引脚是ADC的差分输入通道，各自的前端都有一个可编程增益放大器(PGA)。

一个外部精密电阻将电流转换为电压，然后PGA用最大5倍的增益(分32步)放大该电压信号。目的是补偿检波器二极管响应度的差异，使ADC读取的光功率归一化。外部电阻假定为0.1%精度、5 ppm。一个1 nF电容与该电阻并联，用来抑制宽带噪声。所选电阻值应使得电阻上产生的满量程电压小于 $AV_{DD} - 1.2 V$ 或典型值1.8 V。

PGA设计用于处理最小10 mV输入。为使噪声最低，应旁路ADC输入缓冲器。

当输入电流降为0时，PADC0N由一个缓冲器驱动到0.15 V，以防PGA饱和(参见图15)。

将ADCCON[14]置1可禁用MUX输出上的另一个缓冲器，使得PADCxN引脚也可以连接到接地层。PADCxP引脚同样如此(使用ADCCON[15])。缓冲器在图15中标示的开关旁边。

ADC需要处于伪差分模式，并假定负输入接近地。

所有控制功能都是通过寄存器位独立设置，以赋予用户最大的灵活性。通常，用户需要执行以下步骤：

1. 通过ADCCP和ADCCN寄存器选择PADCxP和PADCxN。
2. 通过ADCCON[15:14]旁路ADC输入缓冲器(可选)。
3. 通过PGA\_GN设置PGA的适当增益值。
4. 通过ADCCON[4:3]将ADC设置为伪差分模式。
5. 启动转换。

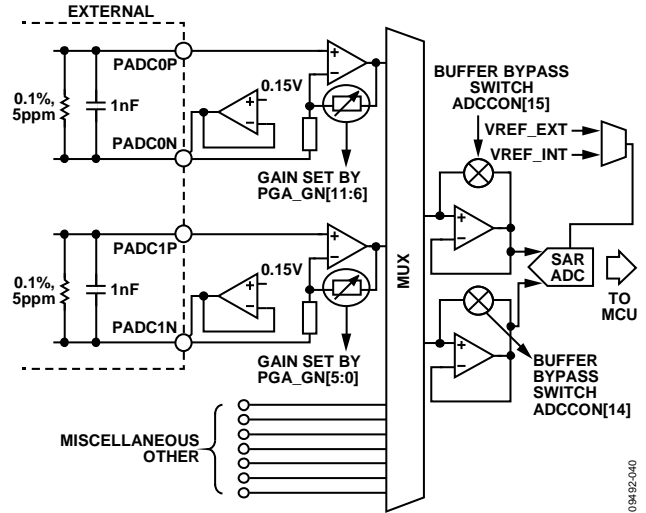


图15. PADC0x/PADC1x ADC输入

## 其他输入通道

ADuC7121包含7个额外的ADC输入引脚。这些引脚也可配置为差分输入对、单端输入或伪差分输入。缓冲器和ADC独立配置，与输入通道选择无关。注意ADC输入缓冲器的输入范围是0.15 V至 $AV_{DD} - 0.15 V$ ；如果输入信号超过此范围，则必须旁路输入缓冲器。

ADuC7121为每路热敏电阻输入提供两个引脚。负输入消除地电压差的误差。选择热敏电阻输入时，务必旁路负端缓冲器，确保放大器不会饱和。配置ADC工作在正伪差分模式下。

除了这些外部输入以外，ADC还可以选择内部输入来监控三个电源：IOVDD、PVDD\_IDAC0和PVDD\_IDAC1。ADC还能监控5路IDAC输出的电压，只需通过寄存器ADCCP选择所需的通道。这些内部信号是单端信号，可通过ADCCN寄存器选择AGND/PGND/IOGND作为ADC的负输入。

注意，当监控IDAC输出、PVDD\_IDAC0、PVDD\_IDAC1或IOVDD\_MON时，必须使能缓冲器，以防干扰ADC采样。

还可以选择片内二极管来监控芯片温度。ADC也可选择 $V_{REF}$ 和AGND作为输入以执行校准。

## PGA和输入缓冲器

PGA是一个单级正增益放大器，支持的输入范围是0.1 V至 $AV_{DD} - 1.2 V$ ，输出摆幅至少应为2.5 V。PGA的增益是从1到5，共有32个线性步进。PADC0x和PADC1x通道的PGA不能旁路。

为确保放大器不发生饱和，PGA的输入电平最大值是 $AVDD - 1.2V$ ，最小值是 $0.1V$ 。输入缓冲器为轨到轨缓冲器，支持 $0.15V$ 至 $AVDD - 0.15V$ 的信号。通过设置ADCCON位[15:14]，正和负输入缓冲器均可独立旁路。

### 典型操作

通过ADC控制和通道选择寄存器配置后，ADC就会转换模拟输入，并在ADC数据寄存器中提供12位结果。

高4位是符号位，12位结果放在位27至位16，如图16所示。同样要注意的是：在全差分模式下，结果表示为二进制补码格式；在伪差分 and 单端模式下，结果表示为标准二进制格式。

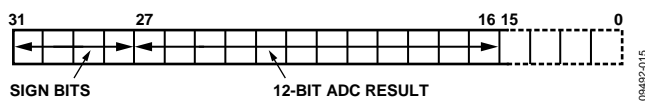


图16. ADC转换结果格式

### 时序

ADC详细时序如图17所示。用户可以控制ADC时钟速度和ADCCON寄存器内采集时钟的数量。默认情况下，采集时间是八个时钟周期，时钟为两分频。附加时钟(如位检验或写入)个数可以设为19，这样采样速率为774 kSPS。对于温度传感器的转换，ADC采集时间自动设置为16时钟，ADC时钟分频器设置为32。当使用包括温度传感器的多通道转换时，在读取温度传感器通道之后，定时设置就会恢复到用户自定义设定。

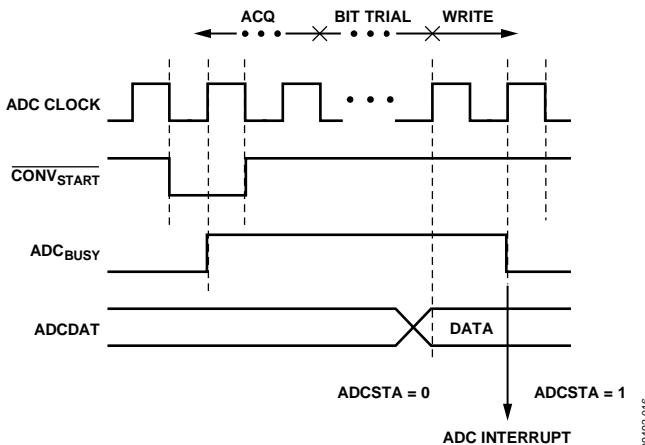


图17. ADC时序

### 温度传感器

ADuC7121提供从片内带隙基准电压源输出并与绝对温度成正比的电压。该电压可以通过前端ADC多路复用器连接到ADC模拟输入通道，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。

内部温度传感器并非设计用作绝对环境温度计算器，而是用作ADuC7121芯片温度的近似指示器。典型温度系数为 $-1.25 mV/^{\circ}C$ 。

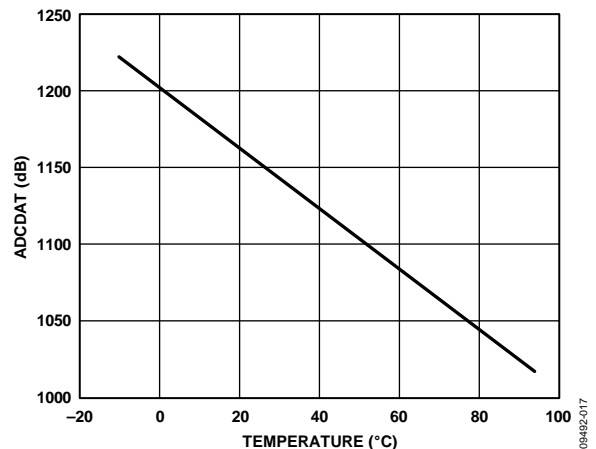


图18. ADC输出与温度的关系

### ADC MMR接口

ADC通过许多MMR进行控制和配置(参见表28)，下面将详细说明这些寄存器。

表28. ADC寄存器

名称	说明
ADCCON	ADC控制寄存器。用户可以通过ADCCON使能ADC外设，选择ADC的工作模式(单端模式、伪差分模式或全差分模式)和转换类型(见表29)。
ADCCP	ADC正向通道选择寄存器。
ADCCN	ADC反向通道选择寄存器。
ADCSTA	ADC状态寄存器。ADCSTA指示ADC转换结果是否就绪。ADCSTA寄存器只有一个位，即ADCREADY(位0)，表示ADC的转换状态。ADC转换结束时，该位置1，并且产生一个ADC中断。当读取ADCDAT寄存器时，该位自动清0。在ADC执行转换时，可以通过外部引脚C3的 $ADC_{BUSY}$ 功能读取ADC的工作状态。在转换期间，该引脚为高电平；当转换结束后， $ADC_{BUSY}$ 变回低电平。如果通过GPOCON寄存器使能，则可以在P0.2引脚输出此信息(参见“通用输入/输出”部分)。
ADCDAT	ADC数据结果寄存器。ADCDAT存放12位ADC转换结果数据，如图16所示。
ADCRST	ADC复位寄存器。ADCRST将所有ADC寄存器复位至默认值。
ADCGN	非PGA通道的ADC增益校准寄存器。
ADCOF	所有ADC通道的ADC失调校准寄存器。
PGA_GN	PGA_PADC0和PGA_PADC1的增益。

# ADuC7121

表29. ADCCON寄存器位功能描述(地址 = 0xFFFF0500, 默认值 = 0x00000A00)

位	值	说明
31:16		这些位保留。
15	0 1	正ADC缓冲器旁路。 用户置0时, 使能正ADC缓冲器。 用户置1时, 旁路正ADC缓冲器。
14	0 1	负ADC缓冲器旁路。 用户置0时, 使能负ADC缓冲器。 用户置1时, 旁路负ADC缓冲器。
13:11	000 001 010 011 100 101	ADC时钟速度。 $f_{ADC} = f_{CORE} \text{ Conversion} = 19 \text{ 时钟} + \text{采集时间}$ $f_{ADC}$ 1分频。该分频器用于在外部时钟频率小于41.78 MHz时获得1 MSPS ADC。 $f_{ADC}$ 2分频(默认值)。 $f_{ADC}$ 4分频。 $f_{ADC}$ 8分频。 $f_{ADC}$ 16分频。 $f_{ADC}$ 32分频。
10:8	000 001 010 011 100 101	ADC采集时间(ADC时钟数)。 2个时钟。 4个时钟。 8个时钟(默认值)。 16个时钟。 32个时钟。 64个时钟。
7		使能转换。 用户置1时, 使能转换模式。 用户清0时, 禁用转换模式。
6		保留。用户将该位置为0。
5	1 0	ADC电源控制。 用户置1时, ADC进入正常模式。ADC上电后必须经过至少5 $\mu$ s才能正确转换。 用户清0时, ADC进入省电模式。
4:3	00 01 10 11	转换模式。 单端模式。 差分模式。 伪差分模式。 保留。
2:0	000 001 010 011 100 101 110 其他	转换类型。 使能引脚F3的ADC <sub>CONVST</sub> 功能作为转换输入。 定时器1输出作为转换使能信号。 定时器0输出作为转换使能信号。 单次软件转换。转换后自动设置为000。 连续软件转换。 PLA转换。 PWM转换。 保留。

表30. ADCCP<sup>1</sup>寄存器位功能描述

位	值	说明
7:5		保留
4:0		正向通道选择位
	00000	PADC0P
	00001	PADC1P
	00010	保留
	00011	保留
	00100	保留
	00101	保留
	00110	ADC4
	00111	ADC5
	01000	ADC6
	01001	ADC7
	01010	ADC8
	01011	ADC9
	01100	ADC10/AINCM
	01101	温度传感器
	01110	DVDD_IDAC0
	01111	DVDD_IDAC1
	10000	DVDD_IDAC2
	10001	DVDD_IDAC3
	10010	DVDD_IDAC4
	10011	IOVDD_MON
	10100	保留
	10101	保留
	10110	V <sub>REF</sub>
	10111	AGND
	Others	保留

<sup>1</sup> ADC通道可用性取决于器件型号。

表31. ADCCN<sup>1</sup>寄存器位功能描述

位	值	说明
7:5		保留
4:0		反向通道选择位
	00000	PADC0N
	00001	PADC1N
	00010	保留
	00011	保留
	00100	保留
	00101	保留
	00110	ADC4
	00111	ADC5
	01000	ADC6
	01001	ADC7
	01010	ADC8
	01011	ADC9
	01100	ADC10/AINCM
	01101	V <sub>REF</sub>
	01110	AGND
	01111	PGND
	10000	IOGND
	Others	保留

<sup>1</sup> ADC通道可用性取决于器件型号。

表32. ADCSTA寄存器位功能描述

位	值	说明
0	1	表示ADC转换完毕。ADC转换完毕后，该位自动置1。
0	0	读取ADCDAT寄存器时，该位自动清0。

表33. ADCDAT寄存器位功能描述

位	值	说明
27:16		保存ADC结果(参见图16)。

表34. ADCRST寄存器位功能描述

位	值	说明
0	1	用户置1时，所有ADC寄存器复位为默认值。

表35. PGA\_GN寄存器位功能描述<sup>1</sup>

位	值 <sup>2</sup>	说明
11:6	不适用	PADC0的PGA增益 = $1 + 4 \times$ (PGA_PADC0_GN/32)。
5:0	不适用	PADC1的PGA增益 = $1 + 4 \times$ (PGA_PADC1_GN/32)。

<sup>1</sup> PGA\_PADC0\_GN和PGA\_PADC1\_GN必须 $\leq 32$ 。

<sup>2</sup> N/A表示不适用。

表36. ADCGN寄存器位功能描述

位	值 <sup>1</sup>	说明
11:6	不适用	这些位保留。
9:0	不适用	非PGA通道的10位ADC增益校准值。

<sup>1</sup> N/A表示不适用。

表37. ADCOF寄存器位功能描述

位	值 <sup>1</sup>	说明
15:10	不适用	这些位保留。
9:0	不适用	10位ADC失调校准值。

<sup>1</sup> N/A表示不适用。

## 转换器操作

ADC集成了一个包含电荷采样输入级的逐次逼近型(SAR)结构。该结构可在三种不同模式下工作：差分模式、伪差分模式、单端模式。

## 差模

ADuC7121包含一个基于两个容性DAC的逐次逼近型ADC。图19和图20分别为ADC采样阶段和转换阶段的简化原理示意图。ADC由控制逻辑、一个SAR和两个容性DAC组成。在信号采样阶段，如图19所示，SW3闭合，SW1和SW2都置于A上，比较器保持在平衡状态，采样电容阵列充电，采集输入端的差分信号。

# ADuC7121

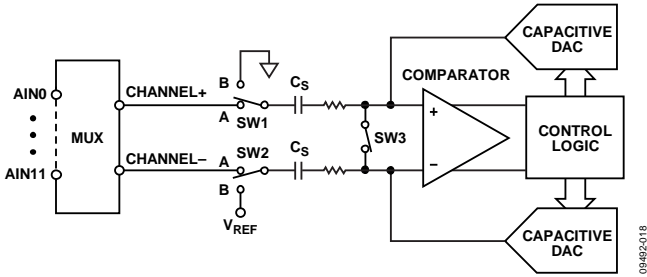


图19. ADC采集阶段

当ADC启动转换(见图20)时, SW3断开, 而SW1和SW2移至位置B, 使得比较器变得不平衡。一旦转换开始, 两个输入均会断开。控制逻辑和电荷再分配DAC可以加上和减去采样电容阵列中的固定电荷数量, 使得比较器恢复到平衡状态。当比较器重新平衡后, 转换就已经完成。

控制逻辑产生ADC的输出代码。注意这里驱动 $V_{IN-}$ 输入和 $V_{IN+}$ 输入的源输出阻抗一定要匹配, 否则由于两个输入的建立时间不同会产生错误。

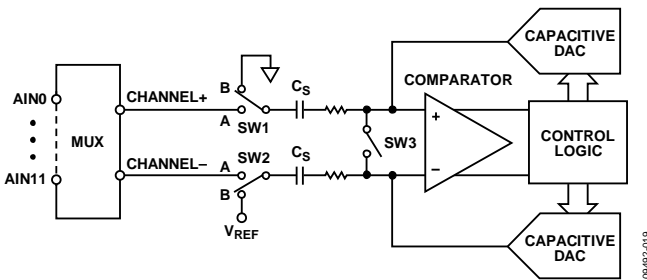


图20. ADC转换阶段

## 伪差分模式

在伪差分模式中, 负通道(Channel-)连接到ADuC7121的 $V_{IN-}$ 输入, SW2开关在A(Channel-)和B( $V_{REF}$ )之间进行切换。 $V_{IN-}$ 输入必须接地或者接一低电压。 $V_{IN+}$ 上的输入信号的范围为 $V_{IN+}$ 至 $V_{REF} + V_{IN-}$ 。注意, 必须恰当选择 $V_{IN-}$ , 不要使 $V_{REF} + V_{IN-}$ 超过 $AV_{DD}$ 。

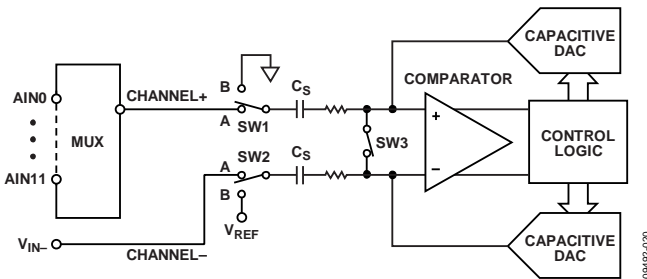


图21. 伪差分模式下ADC

## 单端模式

在单端模式下, SW2始终内部接地。 $V_{IN-}$ 输入可悬空。 $V_{IN+}$ 引脚上的输入信号范围为0 V至 $V_{REF}$ 。

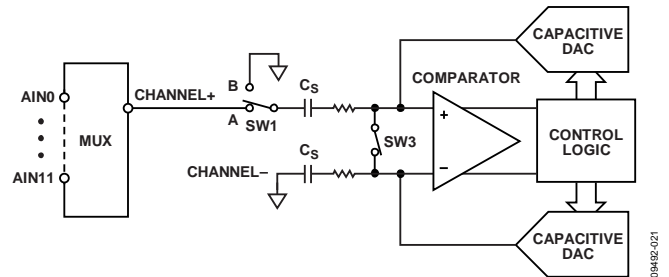


图22. 单端模式下ADC

## 模拟输入结构

ADC模拟输入结构等效电路如图23所示, 图中4个二极管为模拟输入提供ESD保护。切记, 模拟输入信号决不能超过供电轨300 mV以上, 否则会造成这些二极管正偏, 并开始向基板内传导电流。这些二极管可以导通但不会导致器件彻底损坏的最大电流为10 mA。

图23中, 电容C1典型值为4 pF, 可基本上被归属为引脚寄生电容。电阻是由开关阻抗构成的集总元件。电阻典型值为100  $\Omega$ 左右。电容C2为ADC采样电容, 典型值为16 pF。

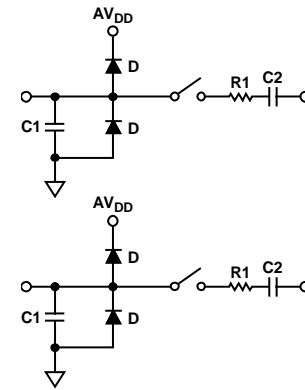


图23. 等效模拟输入电路转换阶段: 开关打开, 采样阶段: 开关关闭

在交流应用中, 建议在相应的模拟输入引脚用一个RC低通滤波器来滤除模拟输入信号的高频成份。在对谐波失真和信噪比要求严格的应用中, 模拟输入应采用一个低阻抗源进行驱动。较大源阻抗会显著影响该ADC的交流性能, 并且可能要求用一个输入缓冲放大器。通常根据具体应用来选择运算放大器。图24和图25为ADC前端的示例。

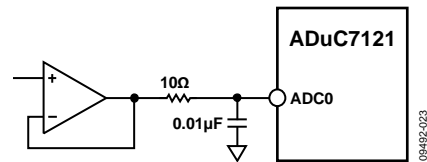


图24. 带缓冲的单端/伪差分输入



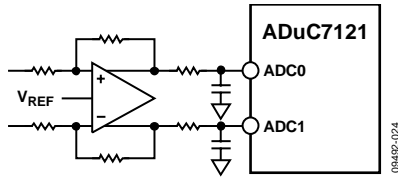


图25. 带缓冲的差分输入

不用放大器来驱动模拟输入端时，应将源阻抗的值限制在1 kΩ以下。源阻抗最大值取决于可容许的总谐波失真(THD)。总谐波失真随着输入源阻抗的增加而增大，从而导致ADC性能下降。

### 驱动模拟输入

该ADC既可以采用内部基准电压源，也可以采用外部基准电压源。差分工作模式对共模输入信号( $V_{CM}$ )有所限制。该限制取决于基准电压值和电源电压，用来确保信号始终处于供电轨以内。

表38给出了一些计算出的 $V_{CM}$ 最小值和最大值。

表38.  $V_{CM}$ 范围

$V_{DD}$	$V_{REF}$	$V_{CM}$ 最小值	$V_{CM}$ 最大值	信号 峰峰值
3.3 V	2.5 V	1.25 V	2.05 V	2.5 V
	2.048 V	1.024 V	2.276 V	2.048 V
	1.25 V	0.75 V	2.55 V	1.25 V
3.0 V	2.5 V	1.25 V	1.75 V	2.5 V
	2.048 V	1.024 V	1.976 V	2.048 V
	1.25 V	0.75 V	2.25 V	1.25 V

表39. REFCON寄存器位功能描述(地址 = 0xFFFF0480, 默认值 = 0x01)

位	说明
7:1	保留。
2	置1时，BUF_VREF1/BUF_VREF2由内部2.5 V基准电压驱动。
1	内部2.5 V基准电压输出使能。 用户置1时，内部2.5 V基准电压源接到 $V_{REF-2.5}$ 引脚。 用户清0时，基准电压源断开与 $V_{REF-2.5}$ 引脚的连接。要连接外部基准电压源， $V_{REF-2.5}$ 引脚也应清除。
0	内部1.2 V基准电压输出使能。 用户置1时，内部1.2 V基准电压源接到 $V_{REF-1.2}$ 引脚。 用户清0时，基准电压源断开与 $V_{REF-1.2}$ 引脚的连接。

### 带隙基准电压源

ADuC7121内置一个可用于ADC和DAC的2.5 V片内带隙基准电压源。此2.5 V基准电压源产生自1.2 V基准电压源。

该内部基准电压也出现在 $V_{REF}$ 引脚( $V_{REF-2.5}$ 和 $V_{REF-1.2}$ )上。当采用内部基准电压源时，各外部 $V_{REF}$ 引脚与AGND之间必须接上一个0.47 μF的电容，以确保ADC转换时的稳定性和快速响应。片内基准电压源也可以连接至外部引脚BUF\_VREF2，作为系统其它电路的基准电压源。

带隙基准电压源也通过缓冲器连接到BUF\_VREF1和BUF\_VREF2引脚。为消除噪声，这些引脚应连接至少0.1 μF电容。带隙基准电压接口包括一个8位REFCON寄存器，如表39所示。

# ADuC7121

## 电源监控器

当IOVDD电源引脚电压降到两个电源跳变点之一以下时，ADuC7121上的电源监控器就会给出提示。监控功能是通过PSMCON寄存器来控制的。在IRQEN或FIQEN寄存器中使能后，监控器使用PSMCON寄存器的PSMI位来中断内核。CMP变为高电平后，该位立即清0。注意，如果在CMP变为高电平(IOVDD电源电压高于跳变点)之前退出所产生的

中断，则在CMP变回高电平之前都不会再次产生中断。用户需要确保在CMP变回高电平之前，代码执行仍在ISR内。

监控功能可以使用户保存当前工作寄存器中的数据，避免由于电压不足或断电造成的数据丢失；它也可以确保直到恢复安全电源时，代码正常重新执行。

使用JTAG调试时，PSM不能正常工作；因此，在JTAG调试模式下应禁用PSM。

**表40. PSMCON寄存器位功能描述(地址 = 0xFFFF0440, 默认值 = 0x0008)**

位	名称	说明
15:4	保留	这些位保留。
3	CMP	比较器位。只读位，可直接反映比较器状态。 该位为1表示IOVDD电压高于所选跳变点或PSM处于掉电模式。 该位为0表示IOVDD电压低于所选跳变点。在退出中断服务程序之前，该位应置1。
2	TP	跳变点选择位。 0 = 2.79 V。 1 = 3.07 V。
1	PSMEN	电源监控器使能位。 用户置1时，使能电源监控器电路。 用户清0时，禁用电源监控器电路。
0	PSMI	电源监控器中断位。CMP变为低电平时，该位会被MicroConverter置1，表明I/O电源电压偏低。PSMI位可用于中断处理器。当CMP变为高电平时，通过对该位写1来清空PSMI位。写0则不影响PSMI位；由于没有超时延迟，所以一旦CMP变为高电平，PSMI位可以立即被清0。

## 非易失性FLASH/EE存储器

### FLASH/EE存储器概述

ADuC7121片内集成了Flash/EE存储器技术，向用户提供非易失、在线可编程存储器空间。

像EEPROM一样，闪存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，通常闪存也被准确地称作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ADuC7121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。

### FLASH/EE存储器

ADuC7121内置两个64 kB的Flash/EE存储器阵列。在模块0中，低62 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，用于将出厂校准系数下载到各种已校准的外设，如带隙基准电压源等。用户代码无法访问这2 kB嵌入式固件。用户无法读取、写入或擦除此页。

闪存模块1中的所有64 kB Flash/EE存储器均可供用户使用。

在串行下载模式或JTAG模式下，可对126 kB的Flash/EE存储器进行在线编程。

### Flash/EE存储器可靠性

ADuC7121上的Flash/EE存储器阵列有两个关键的特性：周期耐久性和数据保持期限。

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、连续的事件，定义如下：

1. 初始页面擦除时序
2. 读/校验时序(单个Flash/EE)
3. 存储器的字节编程时序
4. 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、中和底)的每半个字(16位宽)可循环存取10000次(从0x0000至0xFFFF)。

如“技术规格”部分所示，Flash/EE存储器耐久性是依据JEDEC保持期限规格A117在-10°至+95°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。

保持期限衡量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。这表明每次对Flash/EE存储器进行重新编程时，都保证Flash/EE存储器在完全指定的保持期限内保持数据。注意：基于一个0.6 eV激活能的保持时间随 $T_j$ 递减，如图26所示。

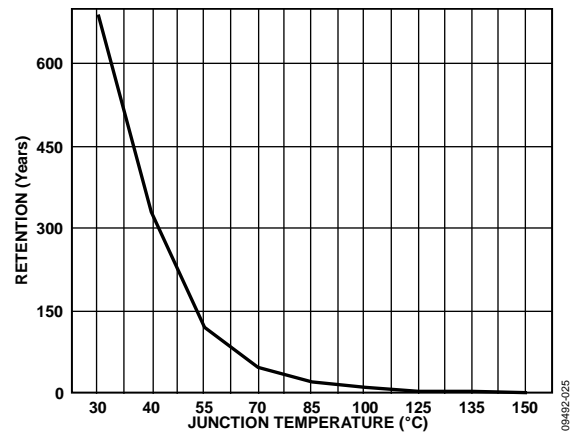


图26. Flash/EE存储器数据保持期限

### 串行下载(在线编程)

ADuC7121可利用I<sup>2</sup>C串行端口下载代码。如果P3.7/BM/PLAO[11]的BM功能通过一个1 kΩ外部电阻下拉到低电平，ADuC7121就会在复位或上电周期后进入到串行下载模式。它与闪存中地址0x00014的状态一起使用。如果此地址为0xFFFFFFFF且BM被拉低，器件就会进入下载模式；如果此地址包含任何其它值，则执行用户代码。在串行下载模式下，当器件在其目标应用硬件中为在线状态时，用户就可以下载代码到Flash/EE存储器的全部126 kB中。开发系统提供PC串行下载可执行文件和硬件适配板，支持通过I<sup>2</sup>C进行串行下载。对于PC下载器，PC最大容许波特率为100 kHz。

### JTAG访问

JTAG协议允许借助片内JTAG接口进行代码下载和调试。

## FLASH/EE存储器安全性

提供给用户的126 kB Flash/EE存储区域实现了读写保护。通过配置FEE0PRO/FEE0HID寄存器的第31位，可以保护模块0的这62 kB，使其不能通过JTAG和I<sup>2</sup>C编程模式来读取。该寄存器的其他31位可以保护Flash/EE存储器不被写入；每一位保护4页，即2 kB。写保护对所有类型的访问都有效。同样，FEE1PRO和FEE1HID保护闪存模块1。通过配置FEE1PRO/FEE1HID寄存器的第31位，可以保护模块1的这64 kB，使其不能通过JTAG读取。位30保护模块1的前8页不被写入。该寄存器的其他30位可以保护Flash/EE存储器不被写入；每一位保护4页，即2 kB。

## 三种级别保护

可通过直接写入FEE<sub>x</sub>HID寄存器来设定或取消保护。复位后，保护失效。

可通过写入FEE<sub>x</sub>PRO寄存器来设定保护。只有在一个保存保护命令(0x0C)和复位以后才生效。FEE<sub>x</sub>PRO寄存器采用一个密钥保护来避免直接访问。该密钥仅保存一次，必须再次输入此密钥才可更改FEE<sub>x</sub>PRO。批量擦除可以将密钥设回0xFFFF，但同时也会擦除所有的用户代码。

通过FEEPRO寄存器和一个特定密钥值(0xDEADDEAD)可以永久保护Flash/EE存储器中的数据。此时即使再次输入密钥也不允许更改FEE<sub>x</sub>PRO寄存器。

## 密钥写入保护寄存器的时序

1. 写入FEE<sub>x</sub>PRO寄存器的某一位，相应页面会被保护；
2. 将FEE<sub>x</sub>MOD寄存器的第6位置1(第5位必须为0)，使能密钥保护；
3. 将一个32位密钥写入FEE<sub>x</sub>ADR和FEE<sub>x</sub>DAT寄存器；
4. 在FEE<sub>x</sub>CON寄存器中运行写密钥命令0x0C，通过监测FEE<sub>x</sub>STA等待读取成功；
5. 复位器件。

要取消或修改保护，可使用相同的序列，同时修改FEE<sub>x</sub>PRO值。如果密钥选择的值是0xDEAD，则无法取消存储器保护。只有批量擦除能够取消器件保护，但它也会擦除所有用户代码。

密钥写入时序如下例所示(保护Flash/EE存储器的第4页到第7页)：

```
FEE0PRO=0xFFFFFFF; //Protect Page 4 to Page 7
FEE0MOD=0x48; //Write key enable
FEE0ADR=0x1234; //16-bit key value
FEE0DAT=0x5678; //16-bit key value
FEE0CON= 0x0C; //Write key command
```

如果要永久保护器件，应该用同样的时序再写一次密钥，但要注意此时FEE<sub>x</sub>ADR = 0xDEAD且FEE<sub>x</sub>DAT = 0xDEAD。

## FLASH/EE控制接口

### FEE0DAT寄存器

FEE0DAT是一个16位数据寄存器。

名称:	FEE0DAT
地址:	0xFFFFF0E0C
默认值:	0xFFFF
访问类型:	读/写

### FEE0ADR寄存器

FEE0ADR是一个16位地址寄存器。

名称:	FEE0ADR
地址:	0xFFFFF0E10
默认值:	0x0000
访问类型:	读/写

### FEE0SGN寄存器

FEE0SGN为24位编码签名。

名称:	FEE0SGN
地址:	0xFFFFF0E18
默认值:	0xFFFFFFFF
访问类型:	只读

### FEE0PRO寄存器

FEE0PRO在随后的寄存器复位后提供保护。它需要一个软件密钥(见表41)。

名称:	FEE0PRO
地址:	0xFFFFF0E1C
默认值:	0x00000000
访问类型:	读/写

### FEE0HID寄存器

FEE0HID提供立即寄存器保护。它不需要任何软件密钥(见表41)。

名称:	FEE0HID
地址:	0xFFFFF0E20
默认值:	0xFFFFFFFF
访问类型:	读/写

表41. FEE0PRO和FEE0HID寄存器位功能描述

位	说明
31	读保护位。 用户清0，保护模块0。 用户置1，允许读取模块0。
30:0	第123页到第0页的写保护。每一位保护4页。 用户清0，在闪存写操作时为这些页面提供写保护。 防止意外写入闪存中的特定页面。 用户置1，允许写入这些页面。

**批量擦除命令执行序列**

```
FEE0DAT = 0x3CFF;
FEE0ADR = 0xFFC3;
FEE0MOD = FEE0MOD|0x8; //Erase key enable
FEE0CON = 0x06; //Mass erase command
```

**FEE1DAT寄存器**

FEE1DAT是一个16位数据寄存器。

名称: FEE1DAT  
地址: 0xFFFF0E8C  
默认值: 0xXXXX  
访问类型: 读/写

**FEE1ADR寄存器**

FEE1ADR是一个16位地址寄存器。

名称: FEE1ADR  
地址: 0xFFFF0E90  
默认值: 0x0000  
访问类型: 读/写

**FEE1SGN寄存器**

FEE1SGN为24位编码签名。

名称: FEE1SGN  
地址: 0xFFFF0E98  
默认值: 0xFFFFF  
访问类型: 只读

**FEE1PRO寄存器**

FEE1PRO在随后的寄存器复位后提供保护。它需要一个软件密钥(见表42)。

名称: FEE1PRO  
地址: 0xFFFF0E9C  
默认值: 0x00000000  
访问类型: 读/写

**FEE1HID寄存器**

FEE1HID提供立即寄存器保护。它不需要任何软件密钥(见表42)。

名称: FEEHID  
地址: 0xFFFF0EA0  
默认值: 0xFFFFFFFF  
访问类型: 读/写

表42. FEE1PRO和FEE1HID寄存器位功能描述

位	说明
31	读保护位。 用户清0，保护模块1。 用户置1，允许读取模块1。
30	第127页到第120页的写保护。 用户清0，在闪存写操作时为这些页面提供写保护。 防止意外写入闪存中的特定页面。 用户置1，允许写入这些页面。
29:0	第119页到第0页的写保护。每一位保护4页。 用户清0，在闪存写操作时为这些页面提供写保护。 防止意外写入闪存中的特定页面。 用户置1，允许写入这些页面。

**FEE0STA寄存器**

名称: FEE0STA  
地址: 0xFFFF0E00  
默认值: 0x0001  
访问类型: 读/写

**FEE1STA寄存器**

名称: FEE1STA  
地址: 0xFFFF0E80  
默认值: 0x0000  
访问类型: 读/写

表43. FEExSTA寄存器位功能描述

位	说明
15:6	保留。
5	保留。
4	保留。
3	Flash/EE中断状态位。 发生中断时，即命令结束且FEExMOD寄存器的Flash/EE中断使能位置1时，自动置1。 读取FEExSTA寄存器后，该位清0。
2	Flash/EE控制器繁忙。 当控制器繁忙时，该位自动置1； 当控制器空闲时，该位自动清0。

# ADuC7121

位	说明
1	命令失败。 当命令的执行失败时，该位自动置1； 读FEEExSTA寄存器后，该位自动清0。
0	命令完成。 当命令完成时，MicroConverter将该位置1。 读FEEExSTA寄存器后，该位自动清0。

## FEE0MOD寄存器

名称： FEE0MOD  
地址： 0xFFFFF0E04  
默认值： 0x80  
访问类型： 读/写

## FEE1MOD寄存器

名称： FEE1MOD  
地址： 0xFFFFF0E84  
默认值： 0x80  
访问类型： 读/写

**表44. FEEExMOD寄存器位功能描述**

位	说明
7:5	保留。除了写密钥时，这些位总是清0。 详情参见“密钥写入保护寄存器的时序”部分。
4	Flash/EE中断使能。 用户将该位置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。 用户将该位置1，使能擦/写命令。 该位清0，防止对Flash/EE存储器执行擦/写命令。
2	保留。用户必须将该位置为0。
1:0	Flash/EE等待状态。为使更改生效，两个Flash/EE模块必须具有相同的等待状态值。

## FEE0CON寄存器

名称： FEE0CON  
地址： 0xFFFFF0E08  
默认值： 0x00  
访问类型： 读/写

## FEE1CON寄存器

名称： FEE1CON  
地址： 0xFFFFF0E88  
默认值： 0x00  
访问类型： 读/写

**表45. FEEExCON中的命令代码**

代码	命令	说明
0x00 <sup>1</sup>	零点	空闲状态。
0x01 <sup>1</sup>	单次读取	将FEEExADR索引的16位数据载入FEEExDAT。
0x02 <sup>1</sup>	单次写入	在FEEExADR指向的地址处写入FEEExDAT。此操作耗时50 μs。
0x03 <sup>1</sup>	擦/写	擦除FEEExADR索引的页并在FEEExADR指向的位置处写入FEEExDAT。此操作耗时20 ms。
0x04 <sup>1</sup>	单次验证	将FEEExADR指向的位置内容和FEEExDAT内的数据进行比较。比较的结果由FEEExSTA的第1位显示。
0x05 <sup>1</sup>	单次擦除	擦除FEEExADR索引的页。
0x06 <sup>1</sup>	批量擦除	擦除用户空间。模块0中的2 kB内核空间是受保护的。此项操作需要2.48 s。为了防止误操作，执行该命令需要一个命令序列。
0x07	保留	保留。
0x08	保留	保留。
0x09	保留	保留。
0x0A	保留	保留。
0x0B	签名	将64 kB Flash/EE的一个签名写入24位FEEExSIGN寄存器中。此项操作需要32,778个时钟周期。
0x0C	保护	此命令只能运行一次。保存FEEExPRO值并只能利用批量擦除(0x06)或密钥加以清除。
0x0D	保留	保留。
0x0E	保留	保留。
0x0F	Ping	无操作，使能中断。

<sup>1</sup> 执行完这些命令后，FEEExCON寄存器总是立即读取0x07。

### SRAM和FLASH/EE执行时间

本部分介绍在执行时间关键的应用中SRAM和Flash/EE的执行访问时间。

#### SRAM执行时间

因为一个最小时钟周期为22 ns，而访问SRAM所需的时间为2 ns，所以从SRAM中取指令只需一个时钟周期。尽管如此，当指令涉及到读取或写入存储器时，若数据在SRAM中时，则需要增加一个时钟周期；若数据在Flash/EE中时，则需要增加三个时钟周期：一个周期用来执行指令，两个周期用来从Flash/EE中检索32位数据。控制流指令(如转移指令)取指令需要一个周期，用新指令填充流水线又需要两个周期。

#### Flash/EE的执行时间

因为Flash/EE为16位，而读取16位字的时间为23 ns，所以从Flash/EE执行指令无法在1个周期内完成(当CD位=0时，从SRAM执行指令就能在一个周期内完成)。此外，无论CD位为何值，在访问数据之前还存在死区时间。

在ARM模式下，指令是32位的，当CD=0时，需要2个时钟周期来读取一个指令；在Thumb模式下，指令是16位的，读取任何指令都只需要1个时钟周期。

当所执行的指令中包含使用Flash/EE进行数据存储的指令时，这两种模式下所用的时间相同。如果指令为控制流指令，除了需要1个额外的时钟周期来解码程序计数器的新地址外，还需要4个时钟周期来填充流水线。如果数据处理指令只涉及内核寄存器，则不需要任何额外的时钟周期，但如果涉及Flash/EE中的数据，则额外需要1个时钟周期来解码数据的地址，然后用2个时钟周期来从Flash/EE中获得32位数据。在取另一个指令之前，需要外加一个时钟周期。对于数据传输指令，执行时间更复杂，具体如表46所示。

表46. ARM/Thumb模式下指令的执行周期

指令	读取周期	死区时间	数据访问	死区时间
LD	2/1	1	2	1
LDH	2/1	1	1	1
LDM/PUSH	2/1	N	2 × N	N
STR	2/1	1	2 × 20 μs	1
STRH	2/1	1	20 μs	1
STRM/POP	2/1	N	2 × N × 20 μs	N

1 < N ≤ 16, N是多重载入/存储指令中欲载入或存储的数据字节数。一条SWAP指令包含LD和STR这两条指令，而且只读取一次，共需8个时钟周期加40 μs。

### 复位和重映射

ARM异常矢量位于存储器阵列的底部，从地址0x00000000到地址0x00000020，如图27所示。

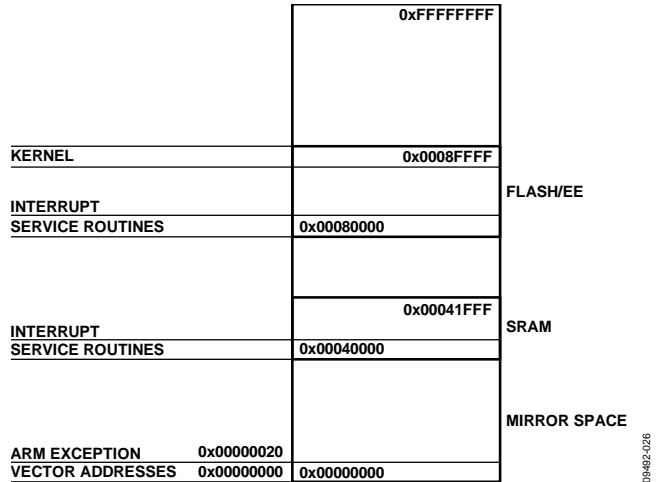


图27. 异常重映射

默认情况下，复位之后，Flash/EE会被镜像到存储器阵列的底部。凭借重映射功能，程序员可将SRAM镜像到存储器阵列的底部，便于从SRAM而不是从Flash/EE执行异常程序。这意味着在32位ARM模式下执行异常程序时，在SRAM中执行的速度要快一倍，因为SRAM的带宽为32位，而Flash/EE带宽为16位。

#### 重映射操作

ADuC7121发生复位时，会自动执行工厂内置的程序代码。该内核是隐藏的，用户代码无法访问。

如果ADuC7121在正常模式工作(P3.7 $\overline{\text{BM}}$ /PLAO[11]引脚为高电平)，则先执行内核的上电配置程序，并跳转到复位矢量地址0x00000000，然后执行用户的复位异常程序。因为复位后，Flash/EE被镜像到存储器阵列的底部，所以复位中断程序一定要写在Flash/EE中。

通过将remap寄存器的位0置1，便可从Flash/EE执行重映射。必须注意从Flash/EE的地址0x00080020上方开始执行此命令，而不能从阵列底部开始执行，因为这部分已定义的存储器空间已被SRAM所取代。

这个操作是可逆的：通过将remap寄存器的位0清0，Flash/EE可以重映射到地址0x00000000。在镜像区域以外的地方执行映射操作时必须小心，任何形式的复位都会把Flash/EE存储器映射到存储器阵列的底部。

# ADuC7121

## 复位操作

一共有四种类型的复位：外部复位、上电复位、看门狗到期复位及软件强制复位。RSTSTA寄存器会指示最近复位的源类型，RSTCLR则可清除RSTSTA寄存器。在复位异常服务程序执行时，可以使用这两种寄存器来识别复位源。如果RSTSTA为空，则为外部复位。注意：当清除RSTSTA时，当前置1的所有位都必须清0，否则将发生复位事件。

RSTCFG寄存器允许不同外设在看门狗复位或软件复位后保持状态不变。

**表47. Remap寄存器位功能描述**  
(地址 = 0xFFFF0220, 默认值 = 0x00)

位	名称	说明
0	Remap	重映射位。 用户将该位置1，将SRAM重映射到0x00000000。 重映射Flash/EE到地址0x00000000后该位自动清0。

**表48. RSTSTA寄存器位功能描述**  
(地址 = 0xFFFF0230, 默认值 = 0x0X)

位	说明
7:3	保留。
2	软件复位。 用户将该位置1，强行使用软件复位； 通过设置对应RSTCLR位清0。
1	看门狗超时。 看门狗定时器超时该位自动置1； 通过设置对应RSTCLR位清0。
0	上电复位。 发生上电复位时自动设置。 通过设置对应RSTCLR位清0。

## RSTCFGKEY0寄存器

名称: RSTCFGKEY0  
地址: 0xFFFF0248  
默认值: 0xXX  
访问类型: 写操作

## RSTCFGKEY1寄存器

名称: RSTCFGKEY1  
地址: 0xFFFF0250  
默认值: 0xXX  
访问类型: 写操作

**表49. RSTCFG写序列**

名称	代码
RSTCFGKEY0	0x76
RSTCFG	用户设定值
RSTCFGKEY1	0xB1

**表50. RSTCFG寄存器位功能描述**  
(地址 = 0xFFFF024C, 默认值 = 0x00)

位	说明
7:4	保留。始终置为0。
3	通过将该位置1，可配置IDAC输出在看门狗复位或软件复位后保持状态不变。 通过将该位清0，IDAC输出引脚和寄存器即会返回其默认状态。
2	通过将该位置1，可配置DAC输出在看门狗复位或软件复位后保持状态不变。 通过将该位清0，DAC输出引脚和寄存器即会返回其默认状态。
1	保留。始终置为0。
0	通过将该位置1，可配置GPIO引脚在看门狗复位或软件复位后保持状态不变。 通过将该位清0，GPIO引脚和寄存器即会返回其默认状态。



## 其他模拟外设

### 数模转换器

ADuC7121片内集成4个12位缓冲电压输出数模转换器(DAC)。每个DAC都有一个轨到轨电压输出缓冲器,驱动能力为5 k $\Omega$ /100 pF。

每个DAC有三种可选范围: 0 V至 $V_{REF}$ (2.5 V内部带隙基准源)、0 V至 $AV_{DD}$ 和0 V至 $EXT\_REF$ (参见图28)。信号范围为0 V

至 $AV_{DD}$ 。注意, DAC也可在插值模式下工作。

### 寄存器接口

每个DAC都可通过一个控制寄存器和一个数据寄存器独立配置。对于这12个DAC来说,这两个寄存器是相同的。本节仅详细说明DAC0CON和DAC0DAT。

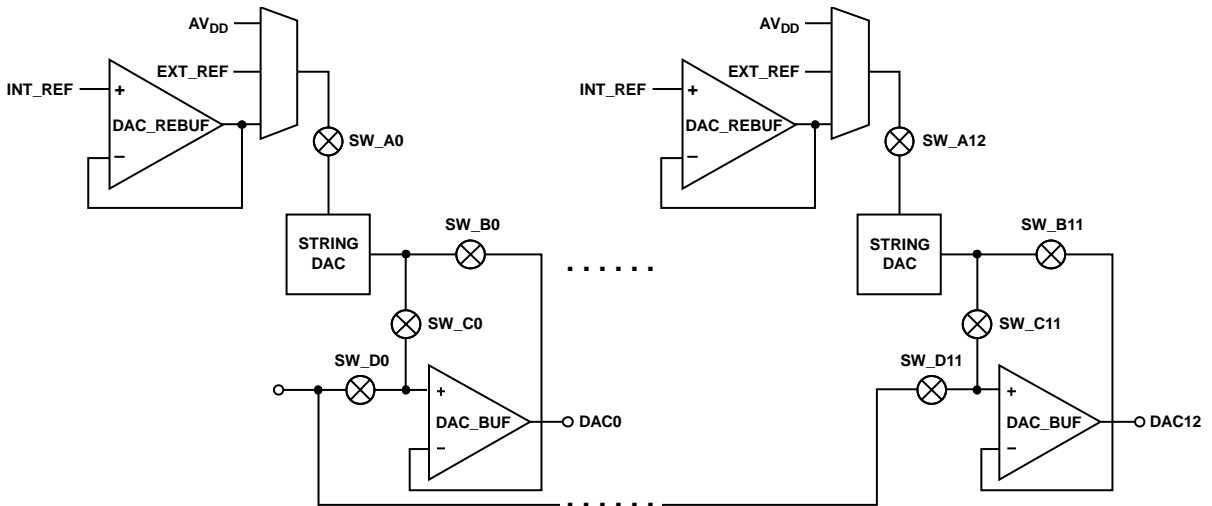


图28. DAC配置

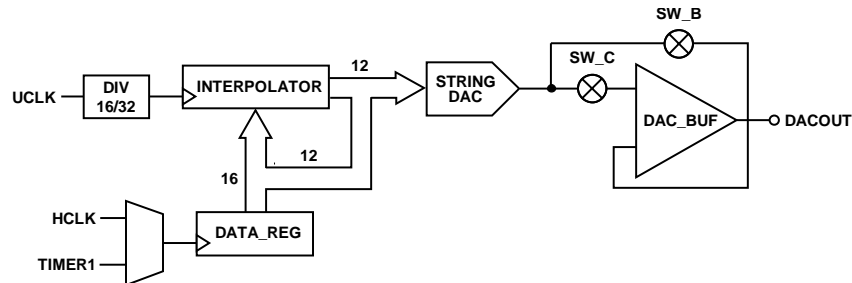


图29. DAC用户功能

# ADuC7121

**表51. DACxCON寄存器(默认值 = 0x100, 读/写访问)**

名称	地址
DAC0CON	0xFFFF0580
DAC1CON	0xFFFF0588
DAC2CON	0xFFFF05B0
DAC3CON	0xFFFF05D8

**表52. DAC0CON寄存器位功能描述**

位	值	名称	说明
15:9	0		保留。
8	1	DACPD	DAC掉电。用户置1时, DACOUTx进入三态模式。
7	0	DACBUF_LP	DAC缓冲器低功耗模式。用户置1时, DAC_BUFF进入低功耗模式。
6	0	BYP	DAC旁路位。 该位置1时, 旁路DAC缓冲器。 清0时, 缓冲DAC输出。
5	0	DACCLK	DAC更新速率。 通过将该位置1, 用户可使用定时器1来更新DAC。 通过将该位清0, 用户可使用HCLK(内核时钟)来更新DAC。
4	0	DACCLR	DAC清除位。 用户置1时, 使能DAC正常工作。 用户清0时, 复位DAC数据寄存器为0。
3	0	Mode	模式位。 用户置1时, DAC工作在正常模式, 并关闭插值器时钟源。用户清0时, 使能插值模式。
2	0	Rate	速率位。用户置1时, 使能插值时钟为HCLK/16。用户清0时, 插值时钟为HCLK/32。
1:0	00 01 10 11	DACRNx	DAC范围位。 DAC范围是从AGND到内部基准电压。 EXT_REF DAC范围是从AGND到外部基准电压。详情参见表39中的REFCON寄存器。 EXT_REF DAC范围是从AGND到外部基准电压。详情参见表39中的REFCON寄存器。 AVDD和AGND。

**表53. DACxDAT寄存器(默认值 = 0x00000000, 读/写访问)**

名称	地址
DAC0DAT	0xFFFF0584
DAC1DAT	0xFFFF058C
DAC2DAT	0xFFFF05B4
DAC3DAT	0xFFFF05DC

**表54. DACxDAT寄存器位功能描述**

位	说明
31:28	保留。
27:16	DACx的12位数据。
15:12	用于插值模式的额外位。
11:0	保留。

## DAC的使用

片内DAC架构由一电阻串DAC和一个输出缓冲放大器构成。功能等效框图如图30所示。

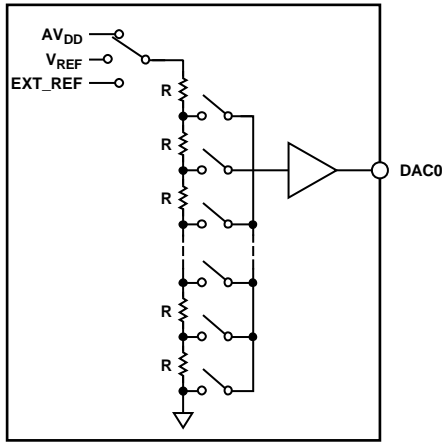


图30. DAC结构

如图30所示，用户可在软件中选择各DAC的基准电压源，可以是 $AV_{DD}$ 、 $V_{REF}$ 或 $EXT\_REF$ 。在0 V至 $AV_{DD}$ 模式中，DAC输出传递函数范围为0 V至 $AV_{DD}$ 引脚电压。在0 V至 $EXT\_REF$ 模式中，DAC输出传递函数范围为0 V至 $V_{REF\_2.5}$ 引脚电压。在0 V至 $V_{REF}$ 模式中，DAC输出传递函数范围为0 V至2.5 V内部基准电压 $V_{REF}$ 。

每个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 $AV_{DD}$ 或地电平的5 mV范围以内。此外，当驱动一个5 k $\Omega$ 阻性负载到地时，除了代码0至100(在0至 $AV_{DD}$ 模式中为代码3995至4095)外，整个传递函数都能保证符合DAC线性度规格要求。

地和 $AV_{DD}$ 附近的线性度下降是由输出放大器的饱和引起的，图31反映了这种效应的一般表现(失调和增益误差忽略不计)。图31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31仅代表输入范围为0 V至 $AV_{DD}$ 模式时的传递函数。在0 V至 $V_{REF}$ 或0 V至 $EXT\_REF$ 模式( $V_{REF} < AV_{DD}$ 或 $EXT\_REF < AV_{DD}$ )，下半部分的非线性度是相似的。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 $V_{REF}$ ，而非 $AV_{DD}$ )，这说明DAC输出没有端点线性误差。

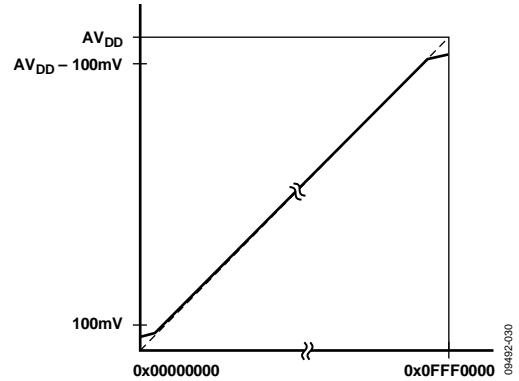


图31. 放大器饱和引起的端点非线性

当有输出负载时，图31所示的端点非线性会变得更差。[ADuC7121](#)数据手册中的技术参数是在DAC输出端接有一个接地的5 k $\Omega$ 阻性负载的条件下得到的。由于DAC输出被强制提供更多的源电流或吸电流，图31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。

## LDO(低压差稳压器)

[ADuC7121](#)集成一个LDO，它从IOVDD电源产生约2.6 V的内核电源电压(DVDD)。由于LDO是从IOVDD驱动，因此IOVDD电源电压需要大于2.7 V。

必须将一个低ESR的0.47  $\mu$ F外部补偿电容(CT)紧挨着各DVDD引脚放置。此电容还用作电荷储存库，供应内核所需的瞬时电荷，特别是在内核时钟(HCLK)的正边沿。

LDO产生的DVDD电压仅用于为[ADuC7121](#)提供电源。因此，用户不应将DVDD引脚用作其他芯片的电源。此外，建议为IOVDD使用良好的电源去耦，以便提高LDO的线性稳压性能。

DVDD引脚无电池反接、限流或热关断保护；因此，在正常工作或电路板制造期间，[ADuC7121](#)的用户任何时候都不得将此引脚短接至地。

## 电流输出DAC (IDAC)

[ADuC7121](#)提供5个电流输出数模转换器(DAC)。电流源(5个电流DAC)可在11位分辨率下提供低噪声和低漂移高端电流输出。5个IDAC具体如下：IDAC0的满量程(FS)输出为250 mA，IDAC1为200 mA，IDAC2为45 mA，IDAC3为80 mA，IDAC4为20 mA。

# ADuC7121

各IDAC的基准电流由内部精密带隙基准电压源和外部精密电阻产生，因此，各IDAC的增益误差受外部电阻的精度影响。外部电阻连接到 $I_{REF}$ 引脚。阻尼电容 $C_{DAMP}$ 限制各IDAC的噪声，选择该电容时应考虑限制噪声带宽并满足信号带宽。 $C_{DAMP}$  IDACx连接到PVDD。

提供一个NMOS开关用于关断IDAC0二极管。注意，当此开关接通时，输出电流关闭。该开关接通时，IDAC0引脚能够承受-0.5 V的电压。上电或复位时，IDAC0默认掉电，其输出为高阻态。使能时，IDAC0输出电流不会过冲。

为了降低芯片热量，可以使用单独的电源。内置LDO可为所有内置低电流IDAC提供稳定的2.5 V电源。

## 精密电流产生和故障保护

基准电流由片内精密带隙电压源或外部基准电压源(默认)产生，然后作用于外部精密电阻。此电阻连接到 $I_{REF}$ 引脚。带隙电压源通过工厂校准获得精密初始值和低温漂移。外部电阻假定为0.1%精度和5 ppm漂移，使用一个0.1  $\mu$ F外部电容来旁路高频噪声。

如果检测到的基准电阻太小，一个故障保护模块会阻止问题发生。将产生的基准电流送至预期外部电阻一半大小的板上电阻，如果该电阻小于预期值的一半(精度约为20%)，则会产生一个故障信号。外部电阻值计算公式为

$R_{EXT} = V_{REF} / I_{REF}$ ，其中 $I_{REF} = 380 \mu A$ ， $V_{REF}$ 是电压转电流电路选定的基准电压。

## IDAC和输出级故障保护

所有5个IDAC利用相同的结构产生高端电流，仅共享产生基准电流的部分。首先利用一个电流模式DAC产生低电流，然后镜像提升至所需的大输出电流。热关断电路防止芯片过热。IDAC保证11位分辨率的单调性。

带宽限制由可编程内部电阻和外部电容实现。目的是滤除高频噪声。另外，它还能从IDAC4的方波输入产生三角波。

当芯片温度超过某一阈值时，热关断电路会自动关断所有输出级。热关断的设计意图是仅在IDAC输出短路时提供保护。其他原因导致的芯片过热也会触发热关断，不过仅IDAC输出级会自动关断。它会触发一个中断，将IDACSTA寄存器中的TSHUT位设置为1，指示芯片过热。

如果数字内核在低于热关断触发点的温度时发生异常，该电路仍能关断IDAC，但必须使用看门狗复位来复位芯片。软件复位或看门狗复位之后，TSHUT位仍会保持其当前值。此位只能通过上电复位、硬件复位或向IDACSTA寄存器写入0的方式清0。

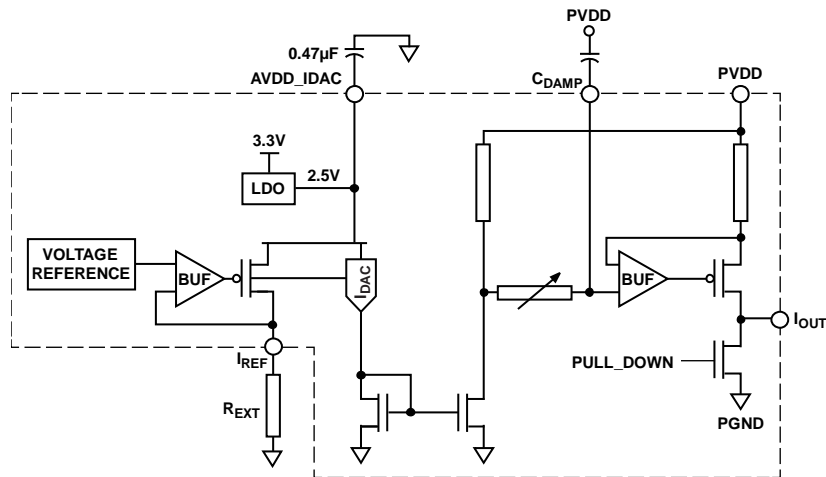


图32.

09492-031

## IDAC寄存器

表55. IDAC控制寄存器(读写访问)

名称	地址(十六进制)	默认值
IDAC0CON	0xFFFF0700	0x0010
IDAC1CON	0xFFFF070C	0x0010
IDAC2CON	0xFFFF0718	0x0010
IDAC3CON	0xFFFF0724	0x0010
IDAC4CON	0xFFFF0730	0x0010
TDSCON	0xFFFF073C	0x00
IDAC0PULLDOWN	0xFFFF0744	0x00

表56. IDACxCON寄存器位功能描述

位	名称	值	说明
15:9			这些位保留。
8:7	SFHMODE	00 01 10 11	位置乱是一种提高IDAC交流精度的方法。请勿用在直流性能非常重要的应用中。 一次置乱递增1。 根据内部计数器置乱。 根据输入数据置乱。 保留。
6	MSBSHFEN	0	MSB置乱使能。 用户置1时, 使能MSB置乱。 用户置0时, 禁用MSB置乱。
5	LSBSHFEN	0	LSB置乱使能。 用户置1时, 使能LSB置乱。 用户置0时, 禁用LSB置乱。
4	IDACPD	1	IDAC掉电位。 用户置1时, IDAC掉电。 IDAC输出为高阻态。 用户置0时, IDAC上电。
3	IDACCLK	0	IDAC更新速率。 通过将该位置1, 用户可使用定时器1来更新IDAC。 通过将该位清0, 用户可使用HCLK(内核时钟)来更新IDAC。
2	IDACCLR	0	IDAC清零位。 用户置1时, 使能IDAC正常工作。 用户清0时, 复位IDAC数据寄存器为0。
1	Mode	0	模式位。此位必须始终为0。
0	保留	0	此位置0。

表57. TDSCON寄存器位功能描述

位	值	名称	说明
7:3		保留	用户将这些位置为0。
2		保留	用户必须将该位置为1。
1	0	DISINT	禁用热触发中断。 用户将该位置0时, 温度超过热关断点就会产生中断。
0	0	DISSD	用户将该位置0时, 温度超过跳变点就会禁用输出电流DAC。

表58. IDAC0PULLDOWN寄存器位功能描述

位	值	名称	说明
7:6		保留	这些位由用户置0。
5	0	Pulldown	IDAC0下拉。 用户置1时, 下拉IDAC0引脚并关断IDAC0。 用户置0时, 禁用下拉。
4	0	PLA_PD_EN	PLA输出触发使能。 用户置1时, 使能PLA输出触发IDAC0下拉。 用户置0时, 禁用此特性。
3:0		PLA Source	PLA输出触发使能的PLA输出源。 用相应的二进制值设置这些位, 就可以选择0到15任意元件的输出。

表59. IDAC数据寄存器  
(默认值 = 0x00000000, 读和写访问)

名称	地址(十六进制)
IDAC0DAT	0xFFFF0704
IDAC1DAT	0xFFFF0710
IDAC2DAT	0xFFFF071C
IDAC3DAT	0xFFFF0728
IDAC4DAT	0xFFFF0734

表60. IDACxDAT寄存器位功能描述

位	名称	值	说明
31:28	保留		这些位保留。
27:17	Data		来自IDACx的数据。
16:0	保留	000	这些位保留。

# ADuC7121

**表61. IDAC带宽寄存器(默认值 = 0x00, 读和写访问)**

名称	地址
IDAC0BW	0xFFFF0708
IDAC1BW	0xFFFF0714
IDAC2BW	0xFFFF0720
IDAC3BW	0xFFFF072C
IDAC4BW	0xFFFF0738

**表62. IDACxBW寄存器位功能描述**

位	名称	值	说明
7:4	保留		用户将这些位置为0。
3:0	BW		带宽控制位。定义RC低通滤波器的3 dB带宽, 假定IDACx的CDAMP_IDACx引脚上有0.01 μF电容。
		000	100 kHz。
		001	28.7 kHz。
		010	15 kHz。
		011	7.8 kHz。
		100	4 kHz。
		101	2.2 kHz。
		110	1.2 kHz。
		其它	未定义。

**表63. IDAC状态寄存器(默认值 = 0x00, 读和写访问)**

名称	地址(十六进制)
IDACSTA	0xFFFF0740

**表64. IDACSTA寄存器位功能描述**

位	值	名称	说明
7:2		保留	这些位由用户置0。
1	0	TSHUT	热关断错误状态位。 内核将该位置1时, 指示发生热关断事件。 内核将该位置0时, 指示IDAC在工作温度范围以内。
0		保留	保留。

## 振荡器和锁相环—电源控制

ADuC7121集成有一个32.768 kHz振荡器、一个时钟分频器和一个锁相环(PLL)。PLL可以锁住多个(1275)内部振荡器, 为系统提供一个稳定的41.78 MHz时钟。为了省电, 内核可以工作在该频率或该频率的二进制约数上。默认的内核时钟为PLL时钟的8分频(CD = 3)或5.2 MHz。内核时钟频率可以通过XCLK引脚输出, 如图33所示。注意, 当使用XCLK引脚输出内核时钟时, 输出信号未经缓冲, 在不使用外部缓冲器的情况下, 不适合用作外部器件的时钟源。

ADuC7121支持掉电模式。

工作模式、时钟模式和可编程时钟分频器通过两个寄存器PLLCON(参见表67)和POWCON(参见表68)进行控制。PLLCON控制时钟系统的工作模式, 而POWCON控制内核时钟频率和掉电模式。

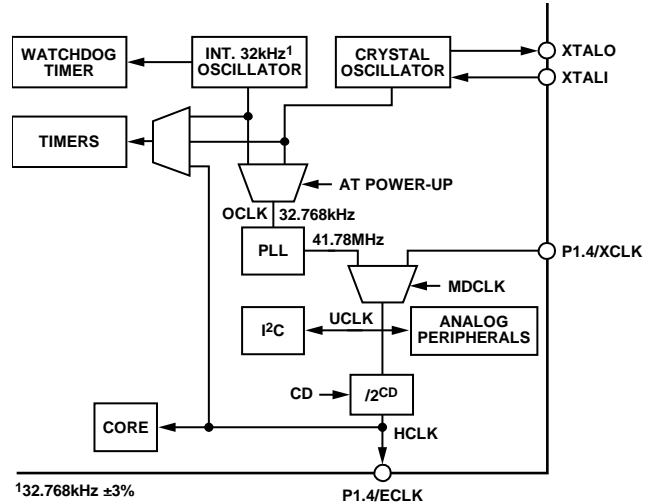


图33. 时钟系统

## 外部晶振选择

要切换到外部晶振, 用户必须执行以下步骤:

1. 使能定时器2中断, 将其超时时间配置为120 μs以上。
2. 执行PLLCON寄存器的写序列, 将MDCLK位设为01, 将OSEL位清0。
3. 执行POWCON寄存器的正确写序列, 强制器件进入浅睡眠(nap)模式。
4. 当定时器2中断源中断器件, 使其离开浅睡眠模式时, 时钟源便切换到外部时钟。

在高噪声环境中, 噪声会耦合到外部晶振引脚, PLL可能会暂时失锁。中断控制器就会出现PLL中断。内核时钟立即挂起, 只有在锁相恢复后才能处理该中断。

为了避免晶振失振, 应该使用看门狗定时器。一般在初始化期间, 应对RSTSTA寄存器进行测试, 判断有没有来自看门狗定时器的复位信号。

**源代码示例**

```

T2LD = 5;
TCON = 0x480;

while ((T2VAL == t2val_old) || (T2VAL > 3))
//ensures timer 值 loaded
    IRQEN = 0x10;
//enable T2 interrupt
    PLLKEY1 = 0xAA;
    PLLCON = 0x01;
    PLLKEY2 = 0x55;

    POWKEY1 = 0x01;
    POWCON =
0x27; // set
core into nap mode
    POWKEY2 = 0xF4;

```

**外部时钟选择**

要切换到P1.4/PWM1/ECLK/XCLK/PLAI[8]引脚的P1.4上的外部时钟，应在模式2下配置P1.4。外部时钟频率最高可达41.78 MHz。

**源代码示例**

```

T2LD = 5;
TCON = 0x480;

while ((T2VAL == t2val_old) || (T2VAL > 3))
//ensures timer 值 loaded
    IRQEN = 0x10;
//enable T2 interrupt
    PLLKEY1 = 0xAA;
    PLLCON = 0x03; //Select external clock
    PLLKEY2 = 0x55;

    POWKEY1 = 0x01;
    POWCON = 0x27; // Set Core into Nap mode
    POWKEY2 = 0xF4;

```

**电源控制系统**

ADuC7121支持多种工作模式选项。

表65给出了不同模式下ADuC7121各部分的上电情况和上电时间。表66给出了不同模式下的总功耗(模拟+数字电源电流)在不同时钟分频器位下的一些典型值。ADC关闭。请注意，这些值也包括测量时稳压器和测试板上其他一些器件的功耗。

**表65. 工作模式**

模式	内核	外设	PLL	XTAL/定时器2/定时器3	外部IRQ	启动/上电时间
有效	开	开	开	开	开	启动/上电时间
暂停		开	开	开	开	24 ns (CD = 0); 3.06 $\mu$ s (CD = 7)
浅休眠			开	开	开	24 ns (CD = 0); 3.06 $\mu$ s (CD = 7)
休眠				开	开	1.58 ms
停止					开	1.7 ms

**表66. 25°C时的典型功耗**

PC[2:0]	模式	CD = 0	CD = 1	CD = 2	CD = 3	CD = 4	CD = 5	CD = 6	CD = 7
000	有效	33.1	21.2	13.8	10	8.1	7.2	6.7	6.45
001	暂停	22.7	13.3	8.5	6.1	4.9	4.3	4	3.85
010	浅休眠	3.8	3.8	3.8	3.8	3.8	3.8	3.8	3.8
011	休眠	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
100	停止	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4

# ADuC7121

## 寄存器和密钥

为了防止意外编程，写入PLLCON和POWCON寄存器时必须遵循特定的顺序(参见表69)。

### PLLKEYx寄存器

名称: PLLKEY1  
地址: 0xFFFF0410  
默认值: 0x0000  
访问类型: 只写

名称: PLLKEY2  
地址: 0xFFFF0418  
默认值: 0x0000  
访问类型: 只写

### PLLCON寄存器

名称: PLLCON  
地址: 0xFFFF0414  
默认值: 0x21  
访问类型: 读/写

表67. PLLCON寄存器位功能描述

位	值	名称	说明
7:6			保留。
5		OSEL	32 kHz PLL输入选择。 用户置1时，使用内部32 kHz振荡器。 默认置1。 用户清0时，使用外部32 kHz晶振。 保留。
4:2			保留。
1:0		MDCLK	时钟模式。
	00		保留。
	01		PLL。默认配置。
	10		保留。
	11		P1.4/PWM1/ECLK/XCLK/PLAI[8]引脚上的外部时钟。

### POWKEYx寄存器

名称: POWKEY1  
地址: 0xFFFF0404  
默认值: 0x0000  
访问类型: 只写

名称: POWKEY2  
地址: 0xFFFF040C  
默认值: 0x0000  
访问类型: 只写

### POWCON寄存器

名称: POWCON  
地址: 0xFFFF0408  
默认值: 0x0003  
访问类型: 读/写

表68. POWCON寄存器位功能描述

位	值	名称	说明
7			保留。
6:4		PC	工作模式。 有效模式。 暂停模式。 浅休眠模式。 休眠模式。IRQ0至IRQ3和定时器2 可以唤醒ADuC7121。 停止模式。 保留。
3		RSVD	保留。
2:0		CD	CPU时钟分频器位。
	000		41.779200 MHz。
	001		20.889600 MHz。
	010		10.444800 MHz。
	011		5.222400 MHz。
	100		2.611200 MHz。
	101		1.305600 MHz。
	110		654.800 kHz。
	111		326.400 kHz。

表69. PLLCON和POWCON写时序

PLLCON	POWCON
PLLKEY1 = 0xAA	POWKEY1 = 0x01
PLLCON = 0x01	POWCON = 用户值
PLLKEY2 = 0x55	POWKEY2 = 0xF4



## 数字外设

### PWM概述

ADuC7121集成了一个6通道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。

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(PWM1和PWM2)示例如图34所示。

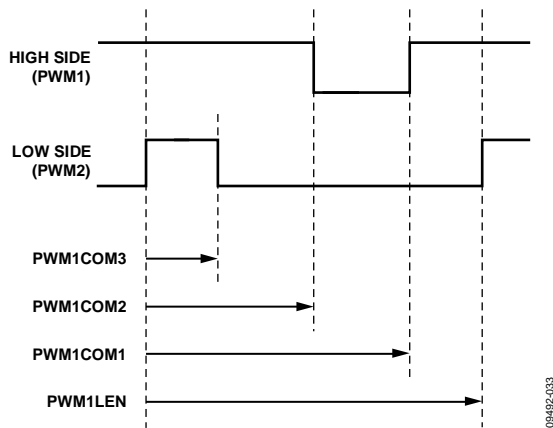


图34. PWM时序

通过PWMCON1，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，如以上PWM1和PWM2的波形所示。

当定时器计数值达到PWM1LEN时，低端波形PWM2变为高电平；当定时器计数值达到PWM1COM3内所保存的数值或者当高端波形PWM1变为低电平时，PWM2变为低电平。

当定时器计数值达到PWM1COM1内所保存的数值时，高端波形PWM1变为高电平；当定时器计数值达到PWM1COM2内所保存的数值时，PWM1变为低电平。

表70. PWM寄存器

名称	功能
PWMCON1	PWM控制
PWM1COM1	用于比较PWM输出1和PWM输出2的比较寄存器1
PWM1COM2	用于比较PWM输出1和PWM输出2的比较寄存器2
PWM1COM3	用于比较PWM输出1和PWM输出2的比较寄存器3
PWM1LEN	用于PWM输出1和PWM输出2的频率控制寄存器
PWM2COM1	用于比较PWM输出3和PWM输出4的比较寄存器1
PWM2COM2	用于比较PWM输出3和PWM输出4的比较寄存器2
PWM2COM3	用于比较PWM输出3和PWM输出4的比较寄存器3
PWM2LEN	用于PWM输出3和PWM输出4的频率控制寄存器
PWM3COM1	用于比较PWM输出5和PWM输出6的比较寄存器1
PWM3COM2	用于比较PWM输出5和PWM输出6的比较寄存器2
PWM3COM3	用于比较PWM输出5和PWM输出6的比较寄存器3
PWM3LEN	用于PWM输出5和PWM输出6的频率控制寄存器
PWMCON2	PWM转换开始控制
PWMICLR	PWM中断清除寄存器

表71. PWMCON1寄存器位功能描述(地址 = 0xFFFF0F80, 默认值 = 0x0012)

位	名称	说明
15	保留	该位为保留位。
14	SYNC	使能PWM同步功能。 用户将该位置1时, 检测到P0.3/MISO/PLAO[12]/SYNC引脚的SYNC上发生由高到低跃迁后, 所有PWM计数器将在下一个时钟沿复位。 用户将该位清0时, 忽略P0.3/MISO/PLAO[12]/SYNC引脚的SYNC上发生的跃迁。
13	PWM6INV	通过将该位置1, 用户可将PWM6反相。 通过将该位清0, 用户可将PWM6置于正常模式下。
12	PWM4INV	通过将该位置1, 用户可将PWM4反相。 通过将该位清0, 用户可将PWM4置于正常模式下。
11	PWM2INV	通过将该位置1, 用户可将PWM2反相。 通过将该位清0, 用户可将PWM2置于正常模式下。
10	PWMTRIP	通过将该位置1, 用户可使能PWM触发中断。当PWMTRIP输入为低电平时, PWMEN位清0, 且产生中断。 通过将该位清0, 用户可禁用PWMTRIP中断。
9	ENA	如果HOFF = 0且HMODE = 1。如果HOFF = 1且HMODE = 1, 请参见表72。如果不是在H桥模式下, 此位无作用。 通过将该位置1, 用户可使能PWM输出。 通过将该位清0, 用户可禁用PWM输出。
8:6	PWMCP[2:0]	PWM时钟预分频器位。设置UCLK分频数。 000 = UCLK 2分频。 001 = UCLK 4分频。 010 = UCLK 8分频。 011 = UCLK 16分频。 100 = UCLK 32分频。 101 = UCLK 64分频。 110 = UCLK 128分频。 111 = UCLK 256分频。
5	POINV	通过将该位置1, 用户可使所有PWM输出反相。 如果用户将该位清0, 则允许正常的PWM输出。
4	HOFF	屏蔽高端。 通过将该位置1, 用户可强制PWM1和PWM3输出高电平信号。这样做还可强制PWM2和PWM4输出低电平信号。 如果用户将该位清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的下一跃迁时, PWMxCOMx的值将被载入内部比较寄存器。 如果用户将该位清0, 则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时, 当PWM3和PWM4处于低电平状态时, 使能PWM1和PWM2作为输出信号。 用户清0时, 当PWM1和PWM2处于低电平状态时, 使能PWM3和PWM4作为输出信号。
1	HMODE	启用H桥模式。 通过将该位置1, 用户可使能H桥模式和PWMCON1的位[5:2]。 通过将该位清0, 用户可让PWM在标准模式下工作。
0	PWMEN	通过将该位置1, 用户可使能全部PWM输出。 通过将该位清0, 用户可禁用全部PWM输出。

在H桥模式下，HMODE = 1，通过表71确定PWM输出，如表72所列。

**表72. PWM输出选择**

PWMCOM1寄存器				PWM输出			
ENA	HOFF	POINV	DIR	PWM1	PWM2	PWMR3	PWM4
0	0	X <sup>1</sup>	X <sup>1</sup>	1	1	1	1
X <sup>1</sup>	1	X <sup>1</sup>	X <sup>1</sup>	1	0	1	0
1	0	0	0	0	0	HS <sup>1</sup>	LS <sup>1</sup>
1	0	0	1	HS <sup>1</sup>	LS <sup>1</sup>	0	0
1	0	1	0	HS <sup>1</sup>	LS <sup>1</sup>	1	1
1	0	1	1	1	1	HS <sup>1</sup>	LS <sup>1</sup>

<sup>1</sup> HS为高端，LS为低端，X为无关位。

上电后，PWMCON1寄存器的默认值为0x12(HOFF = 1且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(参见表73)。

**表73. 比较寄存器(默认值 = 0x0000，读/写访问)**

名称	地址	默认值	访问类型
PWM1COM1	0xFFFF0F84	0x0000	R/W
PWM1COM2	0xFFFF0F88	0x0000	R/W
PWM1COM3	0xFFFF0F8C	0x0000	R/W
PWM2COM1	0xFFFF0F94	0x0000	R/W
PWM2COM2	0xFFFF0F98	0x0000	R/W
PWM2COM3	0xFFFF0F9C	0x0000	R/W
PWM3COM1	0xFFFF0FA4	0x0000	R/W
PWM3COM2	0xFFFF0FA8	0x0000	R/W
PWM3COM3	0xFFFF0FAC	0x0000	R/W

通过向PWMICLR寄存器写入任意值，可以清除PWM触发中断。请注意，使用PWM触发中断时，在退出中断服务程序前，用户应确保PWM中断已被清除。这样可以防止同时产生多个中断。

### PWM转换开始控制

可以配置PWM在有效低端信号变为高电平后产生ADC转换开始信号。低端信号变为高电平到产生转换开始信号之间有一个可编程延迟时间。

该时间通过PWMCON2寄存器控制。如果所选的延迟时间大于PWM脉冲的宽度，中断将保持低电平。

**表74. PWMCON2寄存器位功能描述  
(地址 = 0xFFFF0FB4，默认值 = 0x00)**

位	名称	值	说明
7	CSEN		用户置1时，使能PWM产生转换开始信号。 用户清0时，禁用PWM转换开始信号。
3:0	CSD3 至 CSD0		转换开始延迟。将转换开始信号延迟数个时钟脉冲。  0000 4个时钟脉冲。 0001 8个时钟脉冲。 0010 12个时钟脉冲。 0011 16个时钟脉冲。 0100 20个时钟脉冲。 0101 24个时钟脉冲。 0110 28个时钟脉冲。 0111 32个时钟脉冲。 1000 36个时钟脉冲。 1001 40个时钟脉冲。 1010 44个时钟脉冲。 1011 48个时钟脉冲。 1100 52个时钟脉冲。 1101 56个时钟脉冲。 1110 60个时钟脉冲。 1111 64个时钟脉冲。

计算从转换开始延迟到ADC转换开始的时间时，用户必须考虑内部延迟。下例是延迟4个时钟的情况。需要一个额外时钟来将转换开始信号传送到ADC逻辑。当ADC逻辑收到转换开始信号时，ADC就会在下一ADC时钟沿开始转换(见图35)。

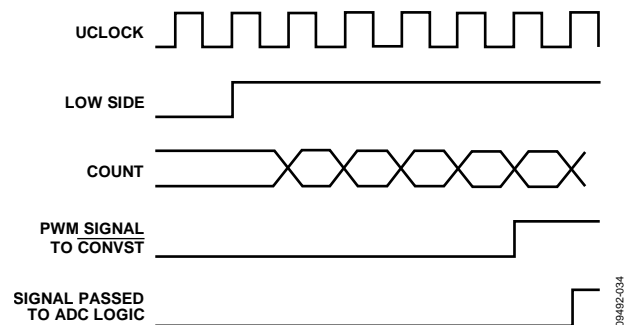


图35. ADC转换

## 通用输入/输出

ADuC7121提供32个通用双向输入/输出(GPIO)引脚。所有I/O引脚都兼容5 V电压，即GPIO支持5 V输入电压。一般来说，许多GPIO引脚都有多种功能(参见表75)。默认情况下，GPIO引脚在GPIO模式下工作。

每个GPIO引脚都有一个内部上拉电阻(约为100 kΩ)，驱动能力为1.6 mA。注意，最多可以有20个驱动1.6 mA电流的GPIO引脚同时工作。32个GPIO被分到4个端口：端口0至端口3。每个端口由4或5个寄存器控制，x表示端口号。

### GPxCON寄存器

名称: GP0CON  
地址: 0xFFFF0D00  
默认值: 0x11000000  
访问类型: 读/写

名称: GP1CON  
地址: 0xFFFF0D04  
默认值: 0x00000000  
访问类型: 读/写

名称: GP2CON  
地址: 0xFFFF0D08  
默认值: 0x00000000  
访问类型: 读/写

名称: GP3CON  
地址: 0xFFFF0D0C  
默认值: 0x00000000  
访问类型: 读/写

GPIO的输入电平在任何时间都能从GPxDAT寄存器中读出，甚至当引脚在除GPIO模式以外的模式下被重新配置时也可以。PLA输入始终有效。

ADuC7121器件进入省电模式后，GPIO引脚维持各自原来的状态。

GPxCON是端口x的控制寄存器，它可以决定端口x中每个引脚的功能。引脚功能如表75所示。

表75. GPIO引脚功能描述

端口	引脚	配置(参见GPxCON, 表76)				
		00	01	10	11	
0	P0.0	GPIO	SCL0	JTAG禁用 JTAG禁用ADC <sub>BUSY</sub> SYNC (PWM) TRIP (PWM) ADC <sub>CONVST</sub>	PLAI[5]	
	P0.1	GPIO	SDA0		PLAI[4]	
	P0.2	GPIO	SPICLK		PLAO[13]	
	P0.3	GPIO	MISO		PLAO[12]	
	P0.4	GPIO	MOSI		PLAI[11]	
	P0.5	GPIO	$\overline{CS}$		PLAI[10]	
	P0.6	GPIO	$\overline{MRST}$		PLAI[2]	
	P0.7	GPIO	$\overline{TRST}$		PLAI[3]	
1	P1.0	GPIO	SIN	SCL1 SDA1	PLAI[7]	
	P1.1	GPIO	SOUT		PLAI[6]	
	P1.2 <sup>1</sup>	TDI (JTAG)	PWM1 PWM2	ECLK/XCLK	PLAO[15]	
	P1.3 <sup>1</sup>	TDO (JTAG)			PLAO[14]	
	P1.4	GPIO			PLAI[8]	
	P1.5	GPIO			PLAI[9]	
	P1.6	GPIO			PLAO[5]	
	P1.7	GPIO			PLAO[4]	
2	P2.0	GPIO/IRQ0	PWM5 PWM6			PLAI[13]
	P2.1	GPIO/IRQ1				PLAI[12]
	P2.2	GPIO		PLAI[1]		
	P2.3	GPIO/IRQ2		PLAI[14]		
	P2.4	GPIO		PLAO[7]		
	P2.5	GPIO		PLAO[6]		
	P2.6	GPIO/IRQ3		PLAI[15]		
	P2.7	GPIO		PLAI[0]		
3	P3.0	GPIO	PWM3 PWM4		PLAO[0]	
	P3.1	GPIO			PLAO[1]	
	P3.2	GPIO/IRQ4			PLAO[2]	
	P3.3	GPIO/IRQ5			PLAO[3]	
	P3.4	GPIO			PLAO[8]	
	P3.5	GPIO			PLAO[9]	
	P3.6	GPIO			PLAO[10]	
	P3.7	GPIO/ $\overline{BM}$			PLAO[11]	

<sup>1</sup> 重新配置这些引脚会禁用JTAG模式。更改默认值后，应擦除器件以重新使能JTAG访问。

**表76. GPxCON寄存器位功能描述**

位	说明
31:30	保留
29:28	选择Px.7引脚的功能
27:26	保留
25:24	选择Px.6引脚的功能
23:22	保留
21:20	选择Px.5引脚的功能
19:18	保留
17:16	选择Px.4引脚的功能
15:14	保留
13:12	选择Px.3引脚的功能
11:10	保留
9:8	选择Px.2引脚的功能
7:6	保留
5:4	选择Px.1引脚的功能
3:2	保留
1:0	选择Px.0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和3的参数。注意：更改GPxPAR寄存器后，必须对GPxDAT寄存器执行写操作。

名称:	GP0PAR
地址:	0xFFFF0D2C
默认值:	0x20000000
访问类型:	读/写
名称:	GP1PAR
地址:	0xFFFF0D3C
默认值:	0x00000000
访问类型:	读/写
名称:	GP2PAR
地址:	0xFFFF0D4C
默认值:	0x00000000
访问类型:	读/写
名称:	GP3PAR
地址:	0xFFFF0D5C
默认值:	0x00222222
访问类型:	读/写

**表77. GPxPAR寄存器位功能描述**

位	说明
31:29	保留
28	Px.7引脚上拉禁用 置1使能上拉 清0禁用上拉
27:25	保留
24	Px.6引脚上拉禁用
23:21	保留
20	Px.5引脚上拉禁用
19:17	保留
16	Px.4引脚上拉禁用
15:13	保留
12	Px.3引脚上拉禁用
11:9	保留
8	Px.2引脚上拉禁用
7:5	保留
4	Px.1引脚上拉禁用
3:1	保留
0	Px.0引脚上拉禁用

## GPxDAT寄存器

GPxDAT是端口x的配置和数据寄存器。它用来配置端口x的GPIO引脚方向，为配置成输出的引脚设置输出值，并为配置成输入的引脚接收和保存输入值。

名称:	GP0DAT
地址:	0xFFFF0D20
默认值:	0x000000XX
访问类型:	读/写
名称:	GP1DAT
地址:	0xFFFF0D30
默认值:	0x000000XX
访问类型:	读/写
名称:	GP2DAT
地址:	0xFFFF0D40
默认值:	0x000000XX
访问类型:	读/写

名称: GP3DAT  
地址: 0xFFFF0D50  
默认值: 0x000000XX  
访问类型: 读/写

表78. GPxDAT寄存器位功能描述

位	说明
31:24	数据传输方向。 用户将该位置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寄存器的数据设置寄存器。

名称: GP0SET  
地址: 0xFFFF0D24  
默认值: 0x000000XX  
访问类型: 只写

名称: GP1SET  
地址: 0xFFFF0D34  
默认值: 0x000000XX  
访问类型: 只写

名称: GP2SET  
地址: 0xFFFF0D44  
默认值: 0x000000XX  
访问类型: 只写

名称: GP3SET  
地址: 0xFFFF0D54  
默认值: 0x000000XX  
访问类型: 只写

表79. GPxSET寄存器位功能描述

位	说明
31:24	保留。
23:16	数据端口x设置位。 用户将该位置1, 则端口x的相应位置1, 同时GPxDAT寄存器中的相应位也置1。 清0不影响数据输出。
15:0	保留。

**GPxCLR寄存器**

GPxCLR为端口x寄存器的数据清除寄存器。

名称: GP0CLR  
地址: 0xFFFF0D28  
默认值: 0x000000XX  
访问类型: 只写

名称: GP1CLR  
地址: 0xFFFF0D38  
默认值: 0x000000XX  
访问类型: 只写

名称: GP2CLR  
地址: 0xFFFF0D48  
默认值: 0x000000XX  
访问类型: 只写

名称: GP3CLR  
地址: 0xFFFF0D58  
默认值: 0x000000XX  
访问类型: 只写

表80. GPxCLR寄存器位功能描述

位	说明
31:24	保留。
23:16	数据端口x清除位。 用户将该位置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。

# ADuC7121

## GPxOCE寄存器

下列GPIO引脚提供开集功能：P1.7、P1.6、端口2和端口3。

**表81. GPxOCE寄存器位功能描述**

位	说明
31:8	保留。
7	GPIO Px.7开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
6	GPIO Px.6开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
5	GPIO Px.5开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
4	GPIO Px.4开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
3	GPIO Px.3开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
2	GPIO Px.2开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
1	GPIO Px.1开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。
0	GPIO Px.0开集使能。 用户置1时，使能开集。 用户置0时，禁用开集。



## UART串行接口

ADuC7121有一个16,450兼容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，以及对从ARM7TDMI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。ADuC7121的P1.0/SIN/SCL1/PLAI[7]和P1.1/SOUT/SDA1/PLAI[6]引脚支持UART功能。

串行通信采用异步协议，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

ADuC7121提供两种UART波特率生成模式：正常450 UART波特率生成模式和ADuC7121小数分频器波特率生成模式。

#### 正常450 UART波特率生成

该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，DL)。标准波特率生成公式为：

$$\text{波特率} = \frac{41.78 \text{ MHz}}{16 \times 2 \times DL} \quad (1)$$

表82列举了几种常见波特率值。

表82.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	0x88	9600	0%
19,200	0x44	19,200	0%
115,200	0x0B	118,691	3%

#### 小数分频器

将小数分频器与正常波特率发生器配合使用，可以生成范围更广、精度更高的波特率。

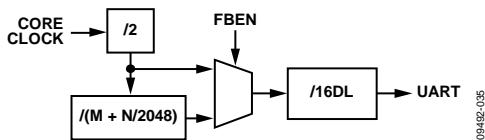


图36.波特率生成选项

采用小数分频器的波特率计算公式如下：

$$\text{波特率} = \frac{41.78 \text{ MHz}}{16 \times DL \times 2 \times \left(M + \frac{N}{2048}\right)} \quad (2)$$

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{\text{波特率} \times 16 \times DL \times 2}$$

例如，生成19,200的波特率，

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{19200 \times 16 \times 67 \times 2}$$

$$M + \frac{N}{2048} = 1.015$$

其中：

$$M = 1$$

$$N = 0.015 \times 2048 = 30$$

$$\text{波特率} = \frac{41.78 \text{ MHz}}{16 \times 67 \times 2 \times \left(1 + \frac{30}{2048}\right)}$$

其中，波特率 = 19,219 bps。

### UART寄存器定义

UART接口包括下列10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。

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的位7为1时，可以访问COMDIV0。

# ADuC7121

## UART发送寄存器

向该8位寄存器写入数据后，允许利用UART发送数据。

名称： COMTX  
地址： 0xFFFF0800  
访问类型： 只写

## UART接收寄存器

对该8位寄存器执行读操作，以接收利用UART发送的数据。

名称： COMRX  
地址： 0xFFFF0800  
默认值： 0x00  
访问类型： 只读

## UART分频锁存器寄存器0

该8位寄存器包含控制UART波特率的分频锁存器的最低有效字节。

名称： COMDIV0  
地址： 0xFFFF0800  
默认值： 0x00  
访问类型： 读/写

## UART分频锁存器寄存器1

该8位寄存器包含控制UART波特率的分频锁存器的最高有效字节。

名称： COMDIV1  
地址： 0xFFFF0804  
默认值： 0x00  
访问类型： 读/写

## UART控制寄存器0

该8位寄存器与COMCON1配合使用，可控制UART操作。

名称： COMCON0  
地址： 0xFFFF080C  
默认值： 0x00  
访问类型： 读/写

表83. COMCON0寄存器位功能描述

位	名称	说明
7	DLAB	分频锁存器访问。 用户置1, 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, 强制发送引脚(SOUT)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, 则置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, 则在字长为5位时发送1.5个停止位, 在字长为6、7或8位时发送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至0	WLS	字长选择。 00 = 5位。 01 = 6位。 10 = 7位。 11 = 8位。

**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。

名称: COMCON1  
地址: 0xFFFF0810  
默认值: 0x00  
访问类型: 读/写

表84. COMCON1寄存器位功能描述

位	名称	说明
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, SOUT被强制处于高电平状态。
3:2		保留位。未使用。
1	RTS	发送请求位。 用户置1, 强制RTS输出为0。 用户清0, 强制RTS输出为1。
0	DTR	数据终端就绪位。 用户置1, 强制DTR输出为0。 用户清0, 强制DTR输出为1。

# ADuC7121

## UART状态寄存器0

名称:	COMSTA0
地址:	0xFFFF0814
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85. COMSTA0寄存器位功能描述**

位	名称	说明
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/SIN/SCL1/PLAI[7]引脚的SIN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中断使能寄存器0**

该8位寄存器用于使能或禁用单独的UART中断源。

名称: COMIEN0  
地址: 0xFFFF0804  
默认值: 0x00  
访问类型: 读/写

**表86. COMIEN0寄存器位功能描述**

位	名称	说明
7至3		保留。未使用。 由用户清0。
2	ELSI	接收引脚(SIN)状态中断使能位。 用户置1, 在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, 当发送期间缓冲为空时, 即当COMSTA[5]置1时使能中断。 由用户清0。
0	ERBFI	使能接收缓冲满中断。 用户置1, 当接收期间缓冲满时使能中断。 由用户清0。

**UART中断识别寄存器0**

该8位寄存器用于反映UART中断源。

名称: COMIID0  
地址: 0xFFFF0808  
默认值: 0x01  
访问类型: 只读

**表87. COMIID0寄存器位功能描述**

状态位 [2:1]	位0 NINT	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

**UART小数分频器寄存器**

该16位寄存器控制ADuC7121小数分频器操作。

名称: COMDIV2  
地址: 0xFFFF082C  
默认值: 0x0000  
访问类型: 读/写

**表88. COMDIV2寄存器位功能描述**

位	名称	说明
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, 利用标准450 UART波特率发生器产生波特率。
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82。
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82。

## I<sup>2</sup>C外设

ADuC7121集成两个I<sup>2</sup>C外设，用户可将这些设备配置成完全I<sup>2</sup>C兼容型总线主机或者完全I<sup>2</sup>C兼容型总线从机。这两种外设完全相同。

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。典型的上拉电阻值介于4.7 kΩ和10 kΩ之间。

用户可对I<sup>2</sup>C总线系统内的I<sup>2</sup>C总线外设地址进行编程。没有进行传输时，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。

I<sup>2</sup>C系统的传输过程为：当总线处于空闲状态时，主机通过产生起始条件来启动传输；在起始地址发送期间，主机发送从机的地址和数据传输方向(R/W)。如果主机没有输掉仲裁且从机进行了应答，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，I<sup>2</sup>C外设无法既作为主机又作为从机。同一个I<sup>2</sup>C通道不能同时支持主机模式和从机模式。ADuC7121的I<sup>2</sup>C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ADuC7121进行编程，可以允许其重复启动一次。在从机模式下，ADuC7121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。
- 在I<sup>2</sup>C主机模式下，ADuC7121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ADuC7121带来任何问题。不过，ADuC7121无法使能时钟延展。
- 在从机模式下，通过对ADuC7121进行编程，可以允许其返回一个非应答信号。这样，可以保证在I<sup>2</sup>C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。
- 在回送模式下，支持内、外部回送以用于I<sup>2</sup>C硬件测试。
- 在主机和从机模式下，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。

### I<sup>2</sup>C功能的外部引脚配置

ADuC7121器件的I<sup>2</sup>C引脚为用于I<sup>2</sup>C0的P0.0和P0.1，以及用于I<sup>2</sup>C1的P1.0和P1.1。P0.0和P1.0的功能是传输I<sup>2</sup>C时钟信号，而P0.1和P1.1的功能是传输I<sup>2</sup>C数据信号。例如，要配置I<sup>2</sup>C0引脚(SCL0和SDA0)，GP0CON寄存器的位0和位4必须置为1，以使能I<sup>2</sup>C模式。而要配置I<sup>2</sup>C1引脚(SCL1和SDA1)，GP1CON寄存器的位1和位5必须置为1，以使能I<sup>2</sup>C模式，如“通用输入/输出部分”所示。

### 串行时钟生成

系统中的I<sup>2</sup>C主机生成传输串行时钟。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 = 0xCF

如果希望串行时钟为400 kHz，那么

应该配置DIVH = 0x28，DIVL = 0x3C

I2CDIV寄存器与DIVH、DIVL相对应。

### I<sup>2</sup>C总线地址 从机模式

在从机模式下，I2CxID0、I2CxID1、I2CxID2和I2CxID3寄存器包含器件ID。器件将4个I2CxIDx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(传输方向位)被忽略。

ADuC7121还支持10位寻址模式。当I2CxSCON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I2CxID0寄存器和I2CxID1寄存器之中。10位地址的组成如下：

I2CxID0[0]：读/写位，不属于I<sup>2</sup>C地址。

- I2CxID0[7:1] = 地址位[6:0]。
- I2CxID1[2:0] = 地址位[9:7]。
- I2CxID1[7:3]的值必须为11110b。

### 主机模式

在主机模式下，I2CxADR0寄存器编程为存储器件的I<sup>2</sup>C地址。

在7位地址模式下，I2CxADR0[7:1]设置为存储器件地址。I2CxADR0[0]是读/写位。

在10位地址模式下，10位地址创建如下：

- I2CxADR0[7:3]必须设置为11110b。
- I2CxADR0[2:1] = 地址位[9:8]。
- I2CxADR1[7:0] = 地址位[7:0]。
- I2CxADR0[0]是读/写位。

**I<sup>2</sup>C寄存器**

I<sup>2</sup>C外设接口由数个寄存器组成，这些寄存器将在以下部分中讨论。

**I<sup>2</sup>C主机寄存器****I<sup>2</sup>C主机控制寄存器**

该16位寄存器用于在主机模式下配置I<sup>2</sup>C外设。

名称： I2C0MCTL, I2C1MCTL

地址： 0xFFFFF0880, 0xFFFFF0900

默认值： 0x0000, 0x0000

访问类型： 读/写

**表89. I2CxMCTL寄存器位功能描述**

位	名称	说明
15:9		保留。这些位为保留位，不允许向其中写入数据。
8	I2CMCENI	I <sup>2</sup> C发送完成中断使能位。 将该位置1，则当从I <sup>2</sup> C总线上检测到停止条件时，使能中断。 清除该中断源。
7	I2CNACKENI	I <sup>2</sup> C非应答信号接收中断使能位。 置1，则当I <sup>2</sup> 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 <sup>2</sup> C仲裁失效中断使能位。 将该位置1，则当I <sup>2</sup> C主机无法获得对I <sup>2</sup> C总线的控制权时，使能中断。 清除该中断源。
5	I2CMTENI	I <sup>2</sup> C发送中断使能位。 将该位置1，则当I <sup>2</sup> C主机完成一个字节的传输后，使能中断。 清除该中断源。
4	I2CMRENI	I <sup>2</sup> C接收中断使能位。 将该位置1，则当I <sup>2</sup> C主机接收到数据时，使能中断。 用户清0，则在I <sup>2</sup> 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 <sup>2</sup> 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，禁用回送模式。
1	I2CBD	I <sup>2</sup> C主机撤回禁用位。 将该位置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 <sup>2</sup> C总线释放后执行撤回。
0	I2CMEN	I <sup>2</sup> C主机使能位。 用户置1，使能I <sup>2</sup> C主机模式。 用户清0，禁用I <sup>2</sup> C主机模式。

# ADuC7121

## I<sup>2</sup>C主机状态寄存器

该16位寄存器是主机模式下的I<sup>2</sup>C状态寄存器。

名称: I2C0MSTA, I2C1MSTA

地址: 0xFFFFF0884, 0xFFFFF0904

默认值: 0x0000, 0x0000

访问类型: 只读

**表90.I2CxMSTA寄存器位功能描述**

位	名称	说明
15:11		保留。这些位保留。
10	I2CBBUSY	I <sup>2</sup> C总线忙碌状态位。 当从I <sup>2</sup> C总线上检测到起始条件后, 该位被置1。 当从I <sup>2</sup> C总线上检测到停止条件后, 该位被清0。
9	I2CMRxFO	主机接收(Rx) 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 <sup>2</sup> 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I2CxMCTL寄存器的I2CMCENI位被置1, 当I2CMTC位的值为1时, 产生中断。 清除该中断源。
7	I2CMNA	I <sup>2</sup> 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I2CxMCTL寄存器的I2CNACKENI位被置1, 当I2CMNA位的值为1时, 产生中断。 在其它条件下, 该位被清0。
6	I2CMBUSY	I <sup>2</sup> 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 <sup>2</sup> C仲裁失效状态位。 当I <sup>2</sup> C主机未获得对I <sup>2</sup> C总线的控制权时, 该位被置1。如果I2CxMCTL寄存器的I2CALENI位被置1, 当I2CAL位的值为1时, 产生中断。 在其它条件下, 该位被清0。
4	I2CMNA	I <sup>2</sup> 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I2CxMCTL寄存器的I2CNACKENI位被置1, 当I2CMNA位的值为1时, 产生中断。 在其它条件下, 该位被清0。
3	I2CMRXQ	I <sup>2</sup> C主机接收请求位。 当数据进入接收FIFO后, 该位被置1。如果I2CxMCTL寄存器的I2CMRENI位被置1, 则产生中断。 在其它条件下, 该位被清0。
2	I2CMTXQ	I <sup>2</sup> C主机发送请求位。 如果发送(Tx) 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I2CxMCTL寄存器的I2CMTENI位被置1, 当I2CMTXQ位的值为1时, 产生中断。 在其它条件下, 该位被清0。
1:0	I2CMTFSTA	I <sup>2</sup> C主机发送FIFO状态位。 00 = I <sup>2</sup> C主机发送FIFO为空。 01 = 主机发送FIFO中包含1个字节的数据。 10 = 主机发送FIFO中包含1个字节的数据。 11 = I <sup>2</sup> C主机发送FIFO已满。



**I<sup>2</sup>C主机接收寄存器**

该8位寄存器是I<sup>2</sup>C主机接收寄存器。

名称: I2C0MRX, I2C1MRX  
地址: 0xFFFF0888, 0xFFFF0908  
默认值: 0x00  
访问类型: 只读

**I<sup>2</sup>C主机发送寄存器**

该8位寄存器是I<sup>2</sup>C主机发送寄存器。

名称: I2C0MTX, I2C1MTX  
地址: 0xFFFF088C, 0xFFFF090C  
默认值: 0x00  
访问类型: 只写

**I<sup>2</sup>C主机读取计数寄存器**

该16位寄存器用于保存主机需要从从机中读取的字节数。

名称: I2C0MCNT0, I2C1MCNT0  
地址: 0xFFFF0890, 0xFFFF0910  
默认值: 0x0000  
访问类型: 读/写

**表91. I2CxMCNT0寄存器位功能描述**

位	名称	说明
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。

**I<sup>2</sup>C主机当前读取计数寄存器**

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。

名称: I2C0MCNT1, I2C1MCNT1  
地址: 0xFFFF0894, 0xFFFF0914  
默认值: 0x00  
访问类型: 只读

**I<sup>2</sup>C地址0寄存器**

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。

名称: I2C0ADR0, I2C1ADR0  
地址: 0xFFFF0898, 0xFFFF0918  
默认值: 0x00  
访问类型: 读/写

**表92. 7位地址模式下的I2CxADR0寄存器**

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

**表93. 10位地址模式下的I2CxADR0寄存器**

位	名称	说明
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, 表示需要读序列。 当该位清0时, 表示需要写序列。

**I<sup>2</sup>C地址1寄存器**

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。

名称: I2C0ADR1, I2C1ADR1  
地址: 0xFFFF089C, 0xFFFF091C  
默认值: 0x00  
访问类型: 读/写

**表94. 10位地址模式下的I2CxADR1寄存器**

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

# ADuC7121

## I<sup>2</sup>C主机时钟控制寄存器

该寄存器用于控制SCL引脚上由主机产生的I<sup>2</sup>C时钟的频率。

名称: I2C0DIV, I2C1DIV

地址: 0xFFFF08A4, 0xFFFF0924

默认值: 0x1F1F

访问类型: 读/写

**表95. I2CxDIV寄存器**

位	名称	说明
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。

## I<sup>2</sup>C从机寄存器

### I<sup>2</sup>C从机控制寄存器

该16位寄存器用于在从机模式下配置I<sup>2</sup>C外设。

名称: I2C0SCTL, I2C1SCTL

地址: 0xFFFF08A8, 0xFFFF0928

默认值: 0x0000

访问类型: 读/写

表96. I2CxSCTL寄存器位功能描述

位	名称	说明
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, 使能中断。 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, 使能中断。 清除该中断源。
8	I2CSSENI	I <sup>2</sup> C停止条件检测中断使能位。 将该位置1, 则当从I <sup>2</sup> C总线上检测到停止条件时, 使能中断。 清除该中断源。
7	I2CNACKEN	I <sup>2</sup> 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 <sup>2</sup> C发送提前中断使能位。 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 <sup>2</sup> C广播状态与ID清除位。 通过向该位写入1, 可清除I2CxSSTA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	I <sup>2</sup> C硬件广播使能位。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, 表明器件接收到一个硬件广播。当器件需要紧急呼叫一个主机而又不知道呼叫哪一个时, 可使用该功能。这是“对可能相关的主机”的呼叫。ADuC7121可监视这些地址。要求主机注意的器件会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, 然后能够处理该器件要求的主机会与其从机通信并进行相应操作。根据2000年1月的I <sup>2</sup> C总线规范, I2CxALT寄存器的LSB应该始终写入1。 通过将该位和I2CGCEN置1, 可在从机模式下使能硬件广播识别。 通过将该位清0, 可禁用硬件广播命令识别。
2	I2CGCEN	I <sup>2</sup> C广播使能位。将该位置1, 可以让从机为I <sup>2</sup> 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 <sup>2</sup> C总线规范, 此时I <sup>2</sup> C接口复位。这个命令可用于复位整个I <sup>2</sup> 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。 通过将该位置1, 可允许从机应答I <sup>2</sup> C广播命令。 通过将该位清0, 可禁用广播命令识别。
1	保留	始终将该位置为0。
0	I2CSEN	I <sup>2</sup> C从机使能位。 用户置1, 使能I <sup>2</sup> C从机模式。 清0, 禁用I <sup>2</sup> C从机模式。

# ADuC7121

## I<sup>2</sup>C从机状态寄存器

这些16位寄存器是从机模式下的I<sup>2</sup>C状态寄存器。

名称: I2C0SSTA, I2C1SSTA  
 地址: 0xFFFFF08AC, 0xFFFFF092C  
 默认值: 0x0000, 0x0000  
 访问类型: 读/写

**表97. I2CxSSTA寄存器位功能描述**

位	名称	说明
15		保留位。
14	I2CSTA	下列情况下，此位置1： 检测到后跟匹配地址的起始条件； 接收到起始字节(0x01)； 已使能广播且接收到广播代码(0x00)。 接收到停止条件后，该位清0。
13	I2CREPS	检测到重复起始条件时，该位置1。 接收到停止条件后，该位清0。读取I2CxSSTA寄存器时，该位也会清0。
12:11	I2CID[1:0]	I <sup>2</sup> C地址匹配寄存器。这些位用于表示哪个I2CxIDx寄存器与接收到的地址相匹配。 [00] = 接收到的地址与I2CxID0相匹配。 [01] = 接收到的地址与I2CxID1相匹配。 [10] = 接收到的地址与I2CxID2相匹配。 [11] = 接收到的地址与I2CxID3相匹配。
10	I2CSS	I <sup>2</sup> C起始后停止条件检测位。 若在起始条件和匹配地址之后检测到停止条件，则该位置1。如果I2CxSCTL寄存器的I2CSSENI位置1，则产生中断。 读该寄存器时，该位清0。
9:8	I2CGCID[1:0]	I <sup>2</sup> 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。 通过向I2CxSCTL的I2CGCCLR位写入1，可清除这些位。注意：无法通过广播复位命令来清除这些位。
7	I2CGC	I <sup>2</sup> 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I2CxALT寄存器的值进行比较。 通过向I2CxSCTL的I2CGCCLR位写入1，可将该位清0。
6	I2CSBUSY	I <sup>2</sup> C从机忙碌状态位。 从机接收到起始条件后，该位置1。 在下列条件下，该位被硬件清0： 接收到的地址不与任何I2CxIDx寄存器匹配。 从机收到停止条件。 重复起始地址不与任何I2CxIDx寄存器匹配。
5	I2CSNA	I <sup>2</sup> 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，该位被置位： 从机因发送FIFO内无数据而返回一非应答信号。 I2CxSCTL寄存器的I2CNACKEN位被置1。 在其它条件下，该位被清0。

位	名称	说明
4	I2CSRxF0	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 <sup>2</sup> C从机接收请求位。 从机的接收FIFO不为空时，该位被置1。I2CxSCTL寄存器的I2CSRXENI位置1时，I2CSRXQ位置1会产生中断。 通过对接收FIFO执行读操作或清除操作，可将该位清0。
2	I2CSTXQ	I <sup>2</sup> 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I2CxSCTL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I2CxSCTL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。 I2CxSCTL寄存器的I2CSTXENI位置1时，I2CSTXQ位置1会产生中断。 在其它条件下，该位被清0。
1	I2CSTFE	I <sup>2</sup> 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，则该位为高电平。在读位操作期间，该位在SCL上升沿置位。 在其它条件下，该位清0。
0	I2CETSTA	I <sup>2</sup> C从机发送提前FIFO状态位。 如果I2CxSCTL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I2CxSCTL寄存器的I2CSETEN位置1，则在写位发送的过程中，SCL正向沿之后，I2CETSTA的值立即被置1。 在一次传输过程中，该位只能被置位一次。 对该位执行读操作后，该位被清0。

# ADuC7121

## I<sup>2</sup>C从机接收寄存器

该8位寄存器是I<sup>2</sup>C从机接收寄存器。

名称: I2C0SRX, I2C1SRX  
地址: 0xFFFFF08B0, 0xFFFFF0930  
默认值: 0x00  
访问类型: 只读

## I<sup>2</sup>C从机发送寄存器

该8位寄存器是I<sup>2</sup>C从机发送寄存器。

名称: I2C0STX, I2C1STX  
地址: 0xFFFFF08B4, 0xFFFFF0934  
默认值: 0x00  
访问类型: 只写

## I<sup>2</sup>C硬件广播识别寄存器

I2CxSCTL寄存器的位3置1时, 此8位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作用了。

名称: I2C0ALT, I2C1ALT  
地址: 0xFFFFF08B8, 0xFFFFF0938  
默认值: 0x00  
访问类型: 读/写

## I<sup>2</sup>C从机ID寄存器

### I2C0IDx寄存器

通过编程, 可在这8个I2C0IDx 8位寄存器内设置从机的I<sup>2</sup>C总线ID。详情见“I<sup>2</sup>C总线地址”部分。

名称: I2C0ID0  
地址: 0xFFFFF08BC  
默认值: 0x00  
访问类型: 读/写

名称: I2C0ID1  
地址: 0xFFFFF08C0  
默认值: 0x00  
访问类型: 读/写

名称: I2C0ID2  
地址: 0xFFFFF08C4  
默认值: 0x00  
访问类型: 读/写

名称: I2C0ID3  
地址: 0xFFFFF08C8  
默认值: 0x00  
访问类型: 读/写

名称: I2C1ID0  
地址: 0xFFFFF093C  
默认值: 0x00  
访问类型: 读/写

名称: I2C1ID1  
地址: 0xFFFFF0940  
默认值: 0x00  
访问类型: 读/写

名称: I2C1ID2  
地址: 0xFFFFF0944  
默认值: 0x00  
访问类型: 读/写

名称: I2C1ID3  
地址: 0xFFFFF0948  
默认值: 0x00  
访问类型: 读/写

**I<sup>2</sup>C公共寄存器****I<sup>2</sup>C FIFO状态寄存器**

这些16位寄存器用于存储在主机和从机模式下接收/发送FIFO的状态。

名称:	I2C0FSTA
地址:	0xFFFF08CC
默认值:	0x0000
访问类型:	读/写
名称:	I2C1FSTA
地址:	0xFFFF094C
默认值:	0x0000
访问类型:	读/写

**表98. I2Cx FSTA寄存器位功能描述**

位	名称	说明
15:10		保留位。
9	I2CFMTX	将该位置1, 可清除主机发送FIFO。
8	I2CFSTX	将该位置1, 可清除从机发送FIFO。
7:6	I2CMRXSTA	I <sup>2</sup> 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 <sup>2</sup> 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	I2CSRXSTA	I <sup>2</sup> 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 <sup>2</sup> 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

ADuC7121片内集成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20 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：P0.3/MISO/PLAO[12]/SYNC、P0.4/MOSI/PLAI[11]/TRIP、P0.2/SPICLK/ADC<sub>BUSY</sub>/PLAO[13]和P0.5/ $\overline{\text{CS}}$ /PLAI[10]/ADC<sub>CONVST</sub>。

### SPI MISO(主机输入、从机输出)引脚

P0.3/MISO/PLAO[12]/SYNC引脚的MISO在主机模式下配置为输入线路，在从机模式下配置为输出线路。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。传送的数据是以字节(8位)为单位的串行数据，MSB优先。

### SPI MOSI(主机输出、从机输入)引脚

P0.4/MOSI/PLAI[11]/TRIP引脚的MOSI在主机模式下配置为输出线路，在从机模式下配置为输入线路。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。传送的数据是以字节(8位)为单位的串行数据，MSB优先。

### SPICLK(串行时钟输入/输出)引脚

主机串行时钟(SPICLK)用于在MOSI SPICLK周期内同步发送和接收数据。所以，每隔8个SPICLK周期，芯片发送/接收一个字节。P0.2/SPICLK/ADC<sub>BUSY</sub>/PLAO[13]引脚在主机模式下配置为输出端，在从机模式下配置为输入端。

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：

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

SPI时钟的最高频率与时钟分频器位无关。

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10 Mbps)。

在主机模式和从机模式下，数据均在SPICLK信号的一个沿发送并另一个沿采样。所以，从机时钟的极性和相位必须与主机配置一致。

### SPI片选输入引脚

在SPI从机模式下，置位P0.5/ $\overline{\text{CS}}$ /PLAI[10]/ADC<sub>CONVST</sub>引脚上的 $\overline{\text{CS}}$ ，可启动一次数据传输。 $\overline{\text{CS}}$ 为低电平有效输入信号。SPI端口随后开始发送和接收8位数据，直到 $\overline{\text{CS}}$ 解除置位以结束传输为止。在从机模式下， $\overline{\text{CS}}$ 总是输入。

在SPI主机模式下， $\overline{\text{CS}}$ 是低电平有效输出信号。传输开始后，它自动置位；传输完成后，它自动解除置位。

### SPI功能的外部引脚配置

ADuC7121的SPI引脚为P0.2至P0.5。

- P0.5/ $\overline{\text{CS}}$ /PLAI[10]/ADC<sub>CONVST</sub>为从机片选引脚。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；传输完成后，该引脚的电平为高电平。
- P0.2/SPICLK/ADC<sub>BUSY</sub>/PLAO[13]为SPICLK引脚。
- P0.3/MISO/PLAO[12]/SYNC为主机输入、从机输出引脚。
- P0.4/MOSI/PLAI[11]/TRIP为主机输出、从机输入引脚。

要配置P0.2至P0.5在SPI模式下工作，参见“通用输入/输出”部分。

### SPI寄存器

下列MMR寄存器用来控制SPI接口：SPISTA、SPIRX、SPITX、SPIDIV和SPICON。

#### SPI状态寄存器

该32位寄存器用于存储主机、从机模式下SPI接口的状态。

名称:	SPISTA
地址:	0xFFFF0A00
默认值:	0x0000
访问类型:	只读



表99. SPISTA寄存器位功能描述

位	名称	说明
15:12		保留位。
11	SPIREX	SPI接收FIFO存在过剩字节。接收FIFO中字节的个数超过由SPICON寄存器中SPIMDE位规定的个数后，该位被置1。 FIFO中字节的个数不超过由SPICON寄存器中SPIMDE位规定的个数时，该位被清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。 Rx 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。 当启动一次发送操作且发送FIFO内没有有效数据时，该位置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。

**SPIRX寄存器**

该8位寄存器是SPI接收寄存器。

名称: SPIRX  
地址: 0xFFFF0A04  
默认值: 0x00  
访问类型: 只读

**SPITX寄存器**

该8位寄存器是SPI发送寄存器。

名称: SPITX  
地址: 0xFFFF0A08  
默认值: 0x00  
访问类型: 只写

**SPIDIV寄存器**

该8位寄存器是SPI波特率选择寄存器。

名称: SPIDIV  
地址: 0xFFFF0A0C  
默认值: 0x00  
访问类型: 读/写

**SPI控制寄存器**

该16位寄存器用于在主机和从机模式下配置SPI外设。

名称: SPICON  
地址: 0xFFFF0A10  
默认值: 0x0000  
访问类型: 读/写

表100. SPICON寄存器位功能描述

位	名称	说明
15:14	SPIMDE	SPI IRQ模式位。这些位用于配置在传输过程中何时发生发送/接收中断。 [00] = 传输完1个字节后，产生发送中断。FIFO接收到一个或以上字节后，产生接收中断。 [01] = 传输完2个字节后，产生发送中断。FIFO接收到两个或以上字节后，产生接收中断。 [10] = 传输完3个字节后，产生发送中断。FIFO接收到1个或以上字节时，产生接收中断。 [11] = 传输完4个字节后，产生发送中断。当接收FIFO已满或存在四个字节时，产生接收中断。
13	SPITFLH	SPI发送FIFO清空使能位。 该位置1时，清空发送FIFO。该位无法自清0；需要一个单次清空操作时，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。 该位清0时，禁用发送FIFO清空。
12	SPIRFLH	SPI接收FIFO清除使能位。 该位置1时，清空接收FIFO。该位无法自清0；需要一个单次清空操作时，应将该位置1。该位置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。如果该位置1且SPITMDE = 0，对接收FIFO执行读操作可以启动一次数据传输。 该位清0时，禁用接收FIFO清空。
11	SPICONT	连续传输使能。 用户将该位置1，连续发送使能。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。P0.5/CS/PLAI[10]/ADC <sub>CONVST</sub> 引脚被置位后，在每一次8位串行传输期间，如发送寄存器不为空，该引脚将始终保持有效。 用户清0，禁用连续发送。每一次传输都是单独的8位串行传输。如果SPITX寄存器中存在有效数据，那么在一个串行时钟停转周期后会重新开始发送数据。
10	SPILP	回送使能位。 用户将该位置1，MISO连接到MOSI，用于测试软件。 用户清0，返回正常模式。
9	SPIOEN	从机MISO输出使能位。 将该位置1，可让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。 若将该位置1，则在发送FIFO无有效数据时，SPI发送0x00。 若将该位清0，则在发送FIFO无有效数据时，SPI发送上次发送的值。
6	SPITMDE	SPI传输和中断模式。 用户置1，向SPITX寄存器写入数据时开始发送。只有当Tx为空时产生中断。 用户清0，读取SPIRX寄存器时开始发送。只有当Rx填满时产生中断。
5	SPILF	LSB优先传输使能位。 用户将该位置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，串行时钟高电平空闲。 用户清0，串行时钟低电平空闲。
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次串行位传输的起始位置。 用户清0，串行时钟脉冲出现在每一个串行位传输的末尾。
1	SPIMEN	主机模式使能位。 用户置1，主机模式使能。 用户清0，从机模式使能。
0	SPIEN	SPI使能位。 用户置1，SPI使能。 用户清0，SPI禁用。

## 可编程逻辑阵列(PLA)

ADuC7121集成有一个完整的可编程逻辑阵列(PLA)，它由两个相互独立但内部连接的PLA模块组成。每一个模块包括8个PLA单元，所以每种器件共有16个PLA单元。

每个PLA单元都包含有一个双输入的查找表，通过配置可以实现任何基于双输入和一个触发器的逻辑输出功能，如图37所示。

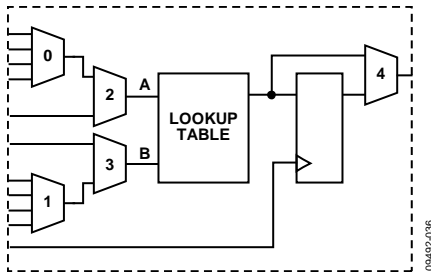


图37. PLA单元

ADuC7121上共有32个GPIO引脚可用于PLA。其中包括16个输入引脚和16个输出引脚，在使用PLA功能之前需要在GPxCON寄存器中对这些引脚进行配置。注意，比较器输出也属于16个输入引脚之一，JTAG TDI和TDO引脚属于PLA输出。若要使用JTAG编程或调试，则不能将JTAG TDI和TDO引脚用作PLA输出。

PLA是通过一组用户寄存器进行配置的。PLA的输出可以连接到内部中断系统、ADC的ADC<sub>CONVST</sub>信号、一个寄存器或者16个PLA输出引脚中的任何一个。

可通过以下方式对两个模块进行互连：

- 单元15(模块1)的输出可以送至单元0(模块0)的多路复用器0的输入0。
- 单元7(模块0)的输出可以送至单元8(模块1)的多路复用器0的输入0。

表101. 单元输入/输出

PLA模块0			PLA模块1		
单元	输入	输出	单元	输入	输出
0	P2.7	P3.0	8	P1.4	P3.4
1	P2.2	P3.1	9	P1.5	P3.5
2	P0.6	P3.2	10	P0.5	P3.6
3	P0.7	P3.3	11	P0.4	P3.7
4	P0.1	P1.7	12	P2.1	P0.3
5	P0.0	P1.6	13	P2.0	P0.2
6	P1.1	P2.5	14	P2.3	P1.3
7	P1.0	P2.4	15	P2.6	P1.2

# ADuC7121

## PLA寄存器接口

PLA外设接口包括21个寄存器，以下是对它们的具体描述。

### PLAELMx寄存器

PLAELMx是单元0到单元15的控制寄存器。通过它们可以配置每一个单元的输入和输出多路复用器、在查找表中选择功能并且选择旁路或使用触发器。参见表103和表106。

**表102. PLAELMx寄存器地址**  
(默认值 = 0x0000, 读/写访问)

名称	地址
PLAELM0	0xFFFF0B00
PLAELM1	0xFFFF0B04
PLAELM2	0xFFFF0B08
PLAELM3	0xFFFF0B0C
PLAELM4	0xFFFF0B10
PLAELM5	0xFFFF0B14
PLAELM6	0xFFFF0B18
PLAELM7	0xFFFF0B1C
PLAELM8	0xFFFF0B20
PLAELM9	0xFFFF0B24
PLAELM10	0xFFFF0B28
PLAELM11	0xFFFF0B2C
PLAELM12	0xFFFF0B30
PLAELM13	0xFFFF0B34
PLAELM14	0xFFFF0B38
PLAELM15	0xFFFF0B3C

**表 103. PLAELMx寄存器位功能描述**

位	值	说明
31:11		保留。
10:9		Mux0控制位(见表106)。
8:7		Mux1控制位(见表106)。
6		Mux2控制位。 用户置1, 选择Mux0的输出。 用户清0, 选择PLADIN中的位值。
5		Mux3控制位。 用户置1, 选择特定单元的输入引脚。 用户清0, 选择Mux1的输出。
4:1	0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111	查找表控制位。 0。 或非。 B与A非。 A非。 A与B非。 B非。 异或。 与非。 与。 同或。 B。 A非或B。 A。 A或B非。 或。 1。
0		Mux4控制位。 用户置1, 旁路触发器。 用户清0, 使用触发器(默认为清0)。

**表104. 反馈配置**

位	值	PLAELM0	PLAELM1至PLAELM7	PLAELM8	PLAELM9至PLAELM15
10:9	00	单元15	单元0	单元7	单元8
	01	单元2	单元2	单元10	单元10
	10	单元4	单元4	单元12	单元12
	11	单元6	单元6	单元14	单元14
8:7	00	单元1	单元1	单元9	单元9
	01	单元3	单元3	单元11	单元11
	10	单元5	单元5	单元13	单元13
	11	单元7	单元7	单元15	单元15

**PLACLK寄存器**

PLACLK是模块0和模块1触发器的时钟选择寄存器。当使用GPIO引脚作为PLA模块的时钟输入时，最大频率为41.78 MHz。

名称: PLACLK  
地址: 0xFFFF0B40  
默认值: 0x00  
访问类型: 读/写

**表105. PLACLK寄存器位功能描述**

位	值	说明
7		保留。
6:4		模块1时钟源选择。
	000	P0.5/ $\overline{\text{CS}}$ /PLAI[10]/ ADC <sub>CONVST</sub> 引脚的P0.5上的GPIO时钟。
	001	P0.0/SCL0/PLAI[5]引脚的P0.0上的GPIO时钟。
	010	P0.7/ $\overline{\text{TRST}}$ /PLAI[3]引脚的P0.7上的GPIO时钟。
	011	HCLK(内核时钟)。
	100	OCLK(32.768 kHz外部晶体)。
	101	定时器1溢出。
	其他	保留。
3		保留。
2:0		模块0时钟源选择。
	000	P0.5/ $\overline{\text{CS}}$ /PLAI[10]/ ADC <sub>CONVST</sub> 引脚的P0.5上的GPIO时钟。
	001	P0.0/SCL0/PLAI[5]引脚的P0.0上的GPIO时钟。
	010	P0.7/ $\overline{\text{TRST}}$ /PLAI[3]引脚的P0.7上的GPIO时钟。
	011	HCLK(内核时钟)。
	100	OCLK(32.768 kHz外部晶体)。
	101	定时器1溢出。
	其他	保留。

**PLAIRQ寄存器**

PLAIRQ用于使能IRQ0和/或IRQ1，以及选择正常中断请求(IRQ)的中断源。

名称: PLAIRQ  
地址: 0xFFFF0B44  
默认值: 0x0000  
访问类型: 读/写

**表106. PLAIRQ寄存器位功能描述**

位	值	说明
15:13		保留。
12		PLA IRQ1使能位。 置1，使能PLA输出IRQ1。 清0，禁用PLA输出IRQ1。

位	值	说明
11:8	0000 0001 1111	PLA IRQ1中断源。 PLA单元0。 PLA单元1。 PLA单元15。
7:5		保留。
4		PLA IRQ0使能位。 置1，使能PLA输出IRQ0。 清0，禁用PLA输出IRQ0。
3:0	0000 0001 1111	PLA IRQ0中断源。 PLA单元0。 PLA单元1。 PLA单元15。

**PLAADC寄存器**

PLAADC是配置PLA作为ADC起始转换信号来源的寄存器。

名称: PLAADC  
地址: 0xFFFF0B48  
默认值: 0x00000000  
访问类型: 读/写

**表107. PLAADC寄存器位功能描述**

位	值	说明
31:5		保留。
4		ADC转换启动使能位。 置1，使能从PLA启动ADC转换。 清0，禁用从PLA启动ADC转换。
3:0	0000 0001 1111	ADC转换起始源选择位。 PLA单元0。 PLA单元1。 PLA单元15。

**PLADIN寄存器**

PLADIN是一个PLA的数据输入寄存器。

名称: PLADIN  
地址: 0xFFFF0B4C  
默认值: 0x00000000  
访问类型: 读/写

**表108. PLADIN寄存器位功能描述**

位	说明
31:16	保留。
15:0	来自单元15至单元0的输入位。

# ADuC7121

## PLADOUT寄存器

PLADOUT是一个PLA的数据输出寄存器。这个寄存器是始终更新的。

名称: PLADOUT  
地址: 0xFFFF0B50  
默认值: 0x00000000  
访问类型: 只读

**表109. PLADOUT寄存器位功能描述**

位	说明
31:16	保留。
15:0	单元15至单元0的输出位。

## PLALCK寄存器

PLALCK是一个PLA锁定选择寄存器。位0只可以写入一次。该位一旦被置1，除了PLADIN寄存器外其它任何PLA寄存器的值都不允许修改。开发系统提供了一套PLA工具，通过它可以很容易地对PLA进行配置。

名称: PLALCK  
地址: 0xFFFF0B54  
默认值: 0x00  
访问类型: 只写

## 中断系统

表 110. IRQ/FIQ 寄存器位功能描述

位	说明	注释
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	定时器0	通用定时器0
3	定时器1	通用定时器1
4	定时器2或唤醒定时器	通用定时器2或唤醒定时器
5	定时器3或看门狗定时器	通用定时器3或看门狗定时器
6	定时器4	通用定时器4
7	IDAC故障	IDAC故障IRQ
8	PSM	电源监控器
9	未定义	该位未使用
10	闪存控制0。	模块0中断的闪存控制器
11	闪存控制1。	模块1中断的闪存控制器
12	模数转换器	ADC中断源位
13	UART	UART中断源位
14	SPI	SPI中断源位
15	I <sup>2</sup> C0主机IRQ	I <sup>2</sup> C主机中断源位
16	I <sup>2</sup> C0从机IRQ	I <sup>2</sup> C从机中断源位
17	I <sup>2</sup> C1主机IRQ	I <sup>2</sup> C主机中断源位
18	I <sup>2</sup> C1从机IRQ	I <sup>2</sup> C从机中断源位
19	XIRQ0(GPIO IRQ0)	外部中断0
20	XIRQ1(GPIO IRQ1)	外部中断1
21	XIRQ2 (GPIO IRQ2)	外部中断2
22	XIRQ3 (GPIO IRQ3)	外部中断3
23	PWM	PWM触发中断源位
24	XIRQ4(GPIO IRQ4)	外部中断4
25	XIRQ5(GPIO IRQ5)	外部中断5
26	PLA IRQ0	PLA模块0 IRQ位
27	PLA IRQ1	PLA模块1 IRQ位

ADuC7121拥有由中断控制器控制的27个中断源。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和FIQ寄存器中的控制位都代表相同的中断源, 如表110所示。

ADuC7121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。

进入中断服务程序(ISR)后, 应立即保存IRQSTA/FIQSTA, 以确保能够响应所有有效中断。

### 正常中断请求(IRQ)

正常中断请求(IRQ)是进入处理器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。器件内有如下4个32位寄存器专门用于IRQ。

### IRQSIG寄存器

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是一个只读寄存器。在中断服务程序中, 不应使用此寄存器来确定IRQ异常的来源, 而只应使用IRQSTA。

# ADuC7121

名称: IRQSIG  
地址: 0xFFFF0004  
默认值: 0x00000000  
访问类型: 只读

## IRQEN寄存器

IRQEN提供当前使能屏蔽的值。将该寄存器的某一位置1, 可使能相应的中断请求, 此时将产生IRQ异常。将某一位清0, 可禁用或屏蔽相应的中断请求, 此时将无法产生IRQ异常。IRQEN寄存器无法用来禁用中断。

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

## IRQCLR寄存器

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。

只能在以下情况下使用此寄存器来禁用中断源:

- 器件处于中断源中断服务程序中。
- 外设被其自己的控制寄存器暂时禁用。

如果IRQ源具有未决中断或可能具有未决中断, 则不应使用IRQCLR来禁用该IRQ源。

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

## IRQSTA寄存器

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。

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。提供此信号的目的是以低延迟处理数据传输或通信通道任务。FIQ接口与IRQ接口相同, 但它提供二级中断(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的位31至位1通过逻辑“或”运算产生FIQ信号到内核以及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使能IRQ和FIQ屏蔽。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会导致FIQEN中的同一位被清0。一个中断源可以同时被IRQEN屏蔽和FIQEN屏蔽禁用。

## FIQSIG寄存器

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。

名称: FIQSIG  
地址: 0xFFFF0104  
默认值: 0x00000000  
访问类型: 只读

## FIQEN寄存器

FIQEN提供了当前使能屏蔽值。将该寄存器的某一位置1, 可使能相应的中断请求, 此时将产生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，会清除FIQEN寄存器的相应位(但不影响其它位)。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。

只能在以下情况下使用此寄存器来禁用中断源：

- 器件处于中断源中断服务程序中。
- 外设被其自己的控制寄存器暂时禁用。

如果FIQ源具有未决中断或可能具有未决中断，则不应使用此寄存器来禁用该FIQ源。

## 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。

## 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。

专用于设置软件中断的32位寄存器为SWICFG，见表111。该寄存器允许控制可编程源中断。

表111. SWICFG寄存器位功能描述

位	说明
31:3	保留。
2	可编程中断FIQ。通过将该位置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IRQ1。通过将该位置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。

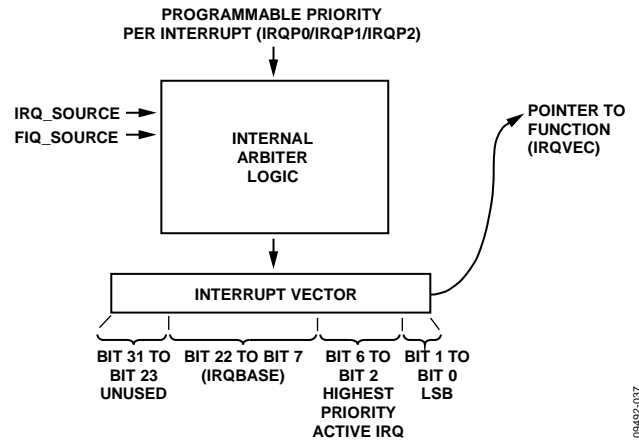


图38. 中断结构

## 矢量中断控制器(VIC)

ADuC7121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IRQP3寄存器，可以为中断源分配不同的中断优先级(0至7)。

# ADuC7121

## VIC寄存器

### IRQBASE寄存器

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。

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

**表112. IRQBASE寄存器位功能描述**

位	类型	初始值	说明
31:16	只读	保留	读数始终为0
15:0	读/写	0	矢量基址

### IRQVEC寄存器

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。仅当产生IRQ中断, 且已通过设置IRQCONN的位0启用IRQ中断嵌套功能后, 才能读取此寄存器。

名称: IRQVEC  
地址: 0xFFFF001C  
默认值: 0x00000000  
访问类型: 读/写

**表113. IRQVEC寄存器位功能描述**

位	类型	初始值	说明
31:23	只读	0	读数始终为0。
22:7	读/写	0	IRQBASE寄存器值。
6:2	只读	0	最高优先级中断源。这些位存储的数据对应0至27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2, 那么这些位的值为[00100]。
1:0	保留	0	保留位。

## 优先级寄存器

### IRQP0寄存器

名称: IRQP0  
地址: 0xFFFF0020  
默认值: 0x00000000  
访问类型: 读/写

**表114. IRQP0寄存器位功能描述**

位	名称	说明
31	保留	保留位。
30:28	IDAC_Fault	为IDAC故障中断设置中断优先级(0至7)。
27	保留	保留位。
26:24	T4PI	为定时器4设置中断优先级(0至7)。
23	保留	保留位。
22:20	T3PI	为定时器3设置中断优先级(0至7)。
19	保留	保留位。
18:16	T2PI	为定时器2设置中断优先级(0至7)。
15	保留	保留位。
14:12	T1PI	为定时器1设置中断优先级(0至7)。
11	保留	保留位。
10:8	T0PI	为定时器0设置中断优先级(0至7)。
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	保留位。

### IRQP1寄存器

名称: IRQP1  
地址: 0xFFFF0024  
默认值: 0x00000000  
访问类型: 读/写

**表115. IRQP1寄存器位功能描述**

位	名称	说明
31	保留	保留位。
30:28	I2COMPI	为I <sup>2</sup> C0主机设置中断优先级(0至7)。
27	保留	保留位。
26:24	SPIPI	为SPI设置中断优先级(0至7)。
23	保留	保留位。
22:20	UARTPI	为UART设置中断优先级(0至7)。
19	保留	保留位。
18:16	ADCPI	为ADC中断源设置中断优先级(0至7)。
15	保留	保留位。
14:12	Flash1PI	为闪存模块1控制器中断源设置中断优先级(0至7)。
11	保留	保留位。
10:8	Flash0PI	为闪存模块0控制器中断源设置中断优先级(0至7)。
7:3	保留	保留位。
2:0	PSMPI	为电源监控器中断源设置中断优先级(0至7)。

**IRQP2寄存器**

名称: IRQP2  
地址: 0xFFFF0028  
默认值: 0x00000000  
访问类型: 读/写

**表116. IRQP2寄存器位功能描述**

位	名称	说明
31	保留	保留位。
30:28	PWMPPI	为PWM设置中断优先级(0至7)。
27	保留	保留位。
26:24	IRQ3PI	为IRQ3设置中断优先级(0至7)。
23	保留	保留位。
22:20	IRQ2PI	为IRQ2设置中断优先级(0至7)。
19	保留	保留位。
18:16	IRQ1PI	为IRQ1设置中断优先级(0至7)。
15	保留	保留位。
14:12	IRQ0PI	为IRQ0设置中断优先级(0至7)。
11	保留	保留位。
10:8	I2C1SPI	为I <sup>2</sup> C1从机设置中断优先级(0至7)。
7	保留	保留位。
6:4	I2C1MPI	为I <sup>2</sup> C1主机设置中断优先级(0至7)。
3	保留	保留位。
2:0	I2C0SPI	为I <sup>2</sup> C0从机设置中断优先级(0至7)。

**IRQP3寄存器**

名称: IRQP3  
地址: 0xFFFF002C  
默认值: 0x00000000  
访问类型: 读/写

**IRQP3寄存器位功能描述**

位	名称	说明
31:15	保留	保留位。
14:12	PLA1PI	为PLA1设置中断优先级(0至7)。
11	保留	保留位。
10:8	PLA0PI	为PLA0设置中断优先级(0至7)。
7	保留	保留位。
6:4	IRQ5PI	为IRQ5设置中断优先级(0至7)。
3	保留	保留位。
2:0	IRQ4PI	为IRQ4设置中断优先级(0至7)。

**IRQCONN寄存器**

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，而第二位用于使能FIQ中断的嵌套和优先级设置。

将这两位清0，则FIQ和IRQ仍可使用，但无法嵌套IRQ或FIQ，也无法设置中断源优先级。在此默认状态下，FIQ的优先级高于IRQ的优先级。

名称: IRQCONN  
地址: 0xFFFF0030  
默认值: 0x00000000  
访问类型: 读/写

**表117. IRQCONN寄存器位功能描述**

位	名称	说明
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寄存器**

如果IRQCONN.0被置位且IRQVEC被读取，则这些位中的一位将被置位。IRQ的优先级决定了第几位将被置位。如果IRQ中断的优先级为0，则位0被置位；如果IRQ中断的优先级为1，则位1被置位；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。

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

**表118. IRQSTAN寄存器位功能描述**

位	名称	说明
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。

# ADuC7121

## FIQVEC寄存器

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。仅当产生FIQ中断, 且已通过设置IRQCONN的位1启用FIQ中断嵌套功能后, 才能读取此寄存器。

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

表119. FIQVEC寄存器位功能描述

位	类型	初始值	说明
31:23	只读	0	读数始终为0。
22:7	读/写	0	IRQBASE寄存器值。
6:2		0	最高优先级中断源。这些位存储的数据对应0至27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2, 那么这些位的值为[00100]。
1:0	保留	0	保留位。

## FIQSTAN寄存器

如果IRQCONN.1被置位且FIQVEC被读取, 则这些位中的一位将被置位。FIQ的优先级决定了第几位将被置位。如果FIQ中断的优先级为0, 则位0被置位; 如果FIQ中断的优先级为1, 则位1被置位; 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

表121. IRQCONE寄存器位功能描述

位	值	名称	说明
31:12		保留	这些位为保留位, 不允许向其中写入数据。
11:10	11 10 01 00	IRQ5SRC[1:0]	外部IRQ5在下降沿触发。 外部IRQ5在上升沿触发。 外部IRQ5在低电平时触发。 外部IRQ5在高电平时触发。
9:8	11 10 01 00	IRQ4SRC[1:0]	外部IRQ4在下降沿触发。 外部IRQ4在上升沿触发。 外部IRQ4在低电平时触发。 外部IRQ4在高电平时触发。
7:6	11 10 01 00	IRQ3SRC[1:0]	外部IRQ3在下降沿触发。 外部IRQ3在上升沿触发。 外部IRQ3在低电平时触发。 外部IRQ3在高电平时触发。

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。

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

表120. FIQSTAN寄存器位功能描述

位	名称	说明
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。

## 外部中断(IRQ0至IRQ5)

ADuC7121可支持6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。而要选择所需的边沿或电平触发, 则必须正确配置IRQCONE寄存器。

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

位	值	名称	说明
5:4	11 10 01 00	IRQ2SRC[1:0]	外部IRQ2在下降沿触发。 外部IRQ2在上升沿触发。 外部IRQ2在低电平时触发。 外部IRQ2在高电平时触发。
3:2	11 10 01 00	IRQ1SRC[1:0]	外部IRQ1在下降沿触发。 外部IRQ1在上升沿触发。 外部IRQ1在低电平时触发。 外部IRQ1在高电平时触发。
1:0	11 10 01 00	IRQ0SRC[1:0]	外部IRQ0在下降沿触发。 外部IRQ0在上升沿触发。 外部IRQ0在低电平时触发。 外部IRQ0在高电平时触发。

**IRQCLRE寄存器**

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

**表122. IRQCLRE寄存器位功能描述**

位	名称	说明
31:25	保留	这些位为保留位，不允许向其中写入数据。
24	IRQ5CLRI	必须在IRQ5中断服务程序中向该位写入1，以清除边沿触发的IRQ5中断。
24	IRQ4CLRI	必须在IRQ4中断服务程序中向该位写入1，以清除边沿触发的IRQ4中断。
23	保留	该位为保留位。
22	IRQ3CLRI	必须在IRQ3中断服务程序中向该位写入1，以清除边沿触发的IRQ3中断。
21	IRQ2CLRI	必须在IRQ2中断服务程序中向该位写入1，以清除边沿触发的IRQ2中断。
20	IRQ1CLRI	必须在IRQ1中断服务程序中向该位写入1，以清除边沿触发的IRQ1中断。
19	IRQ0CLRI	必须在IRQ0中断服务程序中向该位写入1，以清除边沿触发的IRQ0中断。
18:0	保留	这些位为保留位，不允许向其中写入数据。

## 定时器

ADuC7121具有5个通用定时器/计数器。

- 定时器0
- 定时器1
- 定时器2或唤醒定时器
- 定时器3或看门狗定时器
- 定时器4

这5个定时器可以在自由模式或周期模式下工作。

当工作在自由模式时，计数器从最大值/最小值开始递减/递增至0/满量程，然后再从最大值/最小值重新开始递减/递增。

在周期模式下，计数器以载入寄存器(TxLD寄存器)中的值为起始值递减/递增计数至0/满量程，然后再以该寄存器中的值为起始值，重新开始计数。

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(TxCLRI)的清除寄存器内写入任一数据，可以清除IRQ中断。

表123. 事件选择(ES)编号

ES	中断号	名称
00000	2	RTOS定时器(定时器0)
00001	3	GP定时器0(定时器1)
00010	4	唤醒定时器(定时器2)
00011	5	看门狗定时器(定时器3)
00100	6	GP定时器4(定时器4)
00101	7	IDAC故障IRQ
00110	8	电源监控器
00111	9	未定义
01000	10	闪存模块0
01001	11	闪存模块1
01010	12	ADC
01011	13	UART
01100	14	SPI
01101	15	I <sup>2</sup> C0主机
01110	16	I <sup>2</sup> C0从机
01111	17	I <sup>2</sup> C1主机
10000	18	I <sup>2</sup> C1从机
10001	19	外部IRQ0

### 时：分：秒：1/128格式

若要以“时：分：秒：百分数”格式使用定时器，应选择32,768 kHz时钟和预分频值256。百位数字段不代表毫秒，而是一秒的1/128 (256/32,768)。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124。

表124. 时：分：秒：百分数格式

位	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—使用寿命定时器

定时器0是一个带有可编程预分频器的通用48位递增计数或16位递增/递减计数定时器。定时器0采用内核时钟工作，预分频器为1、16、256或32,768。这样，当内核在41.78 MHz频率下工作时，如预分频系数为1，定时器的最小分辨率能够达到22 ns。定时器0也可采用未分频的内核时钟、内部32 kHz振荡器或外部32 kHz晶振工作。

工作在48位模式时，定时器0从0开始递增计数。当前计数器值从T0VAL0和T0VAL1读取。

工作在16位模式时，定时器0既可递增计数，也可递减计数。16位值可通过写入T0LD以载入到计数器内。当前计数器值可从T0VAL0读取。定时器0有一个事件捕获寄存器(T0CAP)，它可以被选定的IRQ中断源初始置位所触发。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。此功能可用于确定事件断言，其精度高于单独通过响应中断的方式。

当定时器0溢出或执行T0ICLR写操作后，定时器0将从T0LD重载数值。

定时器0接口有6个寄存器，如表125所示。

表125. 定时器0接口寄存器

名称	说明
TOLD	该16位寄存器保存计数器的16位载入值。仅适用于16位模式。
TOCAP	该16位寄存器保存已使能IRQ事件所捕捉的16位数值。仅适用于16位模式。
TOVAL0/TOVAL1	TOVAL0是一个16位寄存器，用于保存16个最低有效位(LSB)。 TOVAL1是一个32位寄存器，用于保存32个最高有效位(MSB)。 TOVAL0和TOVAL1为只读寄存器。16位模式使用16位TOVAL0。48位模式同时使用16位TOVAL0和32位TOVAL1。
TOICLR	8位寄存器。向其中写入任意值就可以清除中断。仅适用于16位模式。
TOCON	配置寄存器。

**定时器0值寄存器**

TOVAL0和TOVAL1分别是16位和32位寄存器，保存16个LSB和32个MSB。TOVAL0和TOVAL1为只读寄存器。16位模式使用16位TOVAL0。48位模式同时使用16位TOVAL0和32位TOVAL1。

名称:	TOVAL0
地址:	0xFFFF0304
默认值:	0x0000
访问类型:	只读
名称:	TOVAL1
地址:	0xFFFF0308
默认值:	0x00000000
访问类型:	只读

**定时器0捕捉寄存器**

这是一个16位寄存器，用于保存已使能IRQ事件所捕捉的16位数值；仅适用于16位模式。

名称:	TOCAP
地址:	0xFFFF0314
默认值:	0x0000
访问类型:	只读

**定时器0控制寄存器**

该17位寄存器用于设置定时器0的工作模式。

名称:	TOCON
地址:	0xFFFF030C
默认值:	0x00000000
访问类型:	读/写

表126. TOCON寄存器位功能描述

位	值	说明
31:18		保留。
17		事件选择位。 用户置1，使能对事件的实时捕捉。 用户清0，禁用对事件的实时捕捉。
16:12		事件选择范围是0至16。 事件说明参见“定时器”部分开头。
11		保留。
10:9	00 01 10 11	时钟选择。 内部32 kHz振荡器。 UCLK。 外部32 kHz晶振。 HCLK。
8		递增计数。仅适用于16位模式。 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		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		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)。
5		保留。
4	0 1	定时器0工作模式。 16位工作模式(默认)。 48位工作模式。
3:0	0000 0100 1000 1111	预分频器。 源时钟1分频(默认)。 源时钟16分频。 源时钟256分频。 源时钟32,768分频。

# ADuC7121

## 定时器0载入寄存器

T0LD是一个16位寄存器，用于保存载入计数器的16位值；仅适用于16位模式。

名称： T0LD  
地址： 0xFFFF0300  
默认值： 0x00  
访问类型： 读/写

## 定时器0清除寄存器

通过用户代码向该8位只写寄存器写入任意值，可以刷新(重载)定时器0。

名称： T0CLRI  
地址： 0xFFFF0310  
默认值： 0x00  
访问类型： 只写

## 定时器1—通用定时器

定时器1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。预分频器时钟源可以是32 kHz内部振荡器、32 kHz外部晶振、内核时钟或未分频的PLL时钟输出。时钟源分频系数共有1、16、256或32,768四种。这样，当CD为0且内核在41.78 MHz频率下工作时，如预分频系数为1，定时器的最小分辨率能够达到22 ns。

计数器可以采用标准的32位数格式或者时间格式(时：分：秒：百分数)。

定时器1有一个事件捕获寄存器(T1CAP)，它可以被选定的IRQ中断源初始置位所触发。中断触发时，定时器的当前值被复制到T1CAP内，与此同时，定时器继续运行。此功能可用来更精确地判断事件断言。定时器1接口有5个寄存器，如表127所示。

当器件处于低功耗模式且定时器1从GPIO或低功耗振荡器源接收时钟信号时，定时器1仍继续工作。

当定时器1溢出或执行T1HCLR写操作后，定时器1将从T1LD载入数值。

表127. 定时器1接口寄存器

寄存器	说明
T1LD	32位寄存器。保存32位无符号整数。这是一个只读寄存器。
T1VAL	32位寄存器。保存32位无符号整数。
T1CAP	32位寄存器；保存32位无符号整数。这是一个只读寄存器。
T1CLRI	8位寄存器。向其中写入任意值，可以清除定时器1中断。
T1CON	配置寄存器。

## 定时器1载入寄存器

T1LD是一个32位寄存器，用于保存载入计数器的32位值。

名称： T1LD  
地址： 0xFFFF0320  
默认值： 0x00000000  
访问类型： 读/写

## 定时器1清除寄存器

通过用户代码向该8位只写寄存器写入任意值，可以刷新(重载)定时器1。

名称： T1CLRI  
地址： 0xFFFF032C  
默认值： 0x00  
访问类型： 只写

## 定时器1值寄存器

T1VAL是一个32位寄存器，用于保存定时器1的当前值。

名称： T1VAL  
地址： 0xFFFF0324  
默认值： 0x00000000  
访问类型： 只读

## 定时器1捕捉寄存器

这是一个32位寄存器，用于保存使能IRQ事件所捕捉的32位数值。

名称： T1CAP  
地址： 0xFFFF0330  
默认值： 0x0000  
访问类型： 只读

## 定时器1控制寄存器

该32位寄存器用于设置定时器1的工作模式。

名称： T1CON  
地址： 0xFFFF0328  
默认值： 0x00000000  
访问类型： 读/写



表128. T1CON寄存器位功能描述

位	值	说明
31:24		8位后分频器。
23		使能对后分频器的写操作。
22:20		保留。
19		后分频器比较标志。
18		定时器1中断产生选择标志。
17		事件选择位。 用户置1, 使能对事件的实时捕捉。 用户清0, 禁用对事件的实时捕捉。
16:12		事件选择范围是0至16。事件说明参见“定时器”部分开头。
11:9	000 001 010 011	时钟选择。 内部32 kHz振荡器(默认)。 内核时钟。 UCLK。 P0.6/MRST/PLAI[2]引脚的P0.6。
8		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		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		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	00 01 10 11	格式。 二进制(默认)。 保留。 时:分:秒:百分数: 23小时至0小时。 时:分:秒:百分数: 255小时至0小时。
3:0	0000 0100 1000 1111	预分频器。 源时钟1分频(默认)。 源时钟16分频。 源时钟256分频。 源时钟32,768分频。

## 定时器2—唤醒定时器

定时器2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。预分频器直接从四个时钟源中的一个接收时钟信号：内核时钟(默认选择)、内部32.768 kHz振荡器、外部32.768 kHz时钟晶体或PLL未分频时钟。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41.78 MHz频率下工作时，如预分频系数为1，定时器的最小分辨率能够达到22 ns。如果通过IRQEN[4]使能定时器2，就可以捕捉当前定时器值。

计数器可以采用标准的32位数格式或者时间格式(时：分：秒：百分数)。

当定时器2溢出或执行T2CLR1写操作后，定时器2将从T2LD载入数值。定时器2接口有4个寄存器，如表129所示。

**表129. 定时器2接口寄存器**

寄存器	说明
T2LD	32位寄存器。保存32位无符号整数。
T2VAL	32位寄存器。保存32位无符号整数。这是一个只读寄存器。
T2CLR1	8位寄存器。向其中写入任意值就可以清除定时器2中断。
T2CON	配置寄存器。

## 定时器2载入寄存器

T2LD是一个32位寄存器，用于保存载入计数器的32位值。

名称： T2LD  
地址： 0xFFFF0340  
默认值： 0x00000000  
访问类型： 读/写

## 定时器2清除寄存器

通过用户代码向该8位只写寄存器写入任意值，可以刷新(重载)定时器2。

名称： T2CLR1  
地址： 0xFFFF034C  
默认值： 0x00  
访问类型： 只写

## 定时器2值寄存器

T2VAL为32位寄存器，用于保存定时器2的当前值。

名称： T2VAL  
地址： 0xFFFF0344  
默认值： 0x00000000  
访问类型： 只读

## 定时器2控制寄存器

该32位寄存器用于设置定时器2的工作模式。

名称： T2CON  
地址： 0xFFFF0348  
默认值： 0x00000000  
访问类型： 读/写

**表130. T2CON寄存器位功能描述**

位	值	说明
31:11		保留。
10:9	00 01 10 11	时钟源选择。 内部32.768 kHz振荡器(默认)。 内核时钟。 外部32.768 kHz时钟晶体。 UCLK。
8		递增计数。 用户置1，定时器2递增计数。 用户清0，定时器2递减计数(默认)。
7		定时器2使能位。 用户置1，使能定时器2。 用户清0，禁用定时器2(默认)。
6		定时器2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	00 01 10 11	格式。 二进制(默认)。 保留。 时：分：秒：百分数：23小时至0小时。 时：分：秒：百分数：255小时至0小时。
3:0	0000 0100 1000 1111	预分频器。 源时钟1分频(默认)。 源时钟16分频。 源时钟256分频。 (此设置应与定时器2格式1,0和格式1,1一起使用。) 源时钟32,768分频。

## 定时器3—看门狗定时器

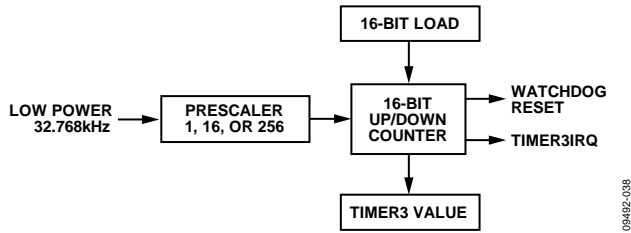


图39. 定时器3框图

定时器3共有两种工作模式：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。

当定时器3溢出或执行T3CLR1写操作后，定时器3将从T3LD载入数值。

### 正常模式

在正常模式下的定时器3和在16位工作模式下的定时器0相同，但时钟源除外。时钟源采用32.768 kHz振荡器，分频系数共有1、16和256三种。定时器3也有捕捉功能，如果通过IRQEN[5]使能定时器2，就可以捕捉当前定时器值。

### 看门狗模式

看门狗模式通过设置T3CON[5]进入。定时器3从T3LD寄存器中的超时值开始递减计数，直至计数值为0。当使用1/256的最大预分频和T3LD的满量程值时，最长超时时间为512秒。

由于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。

T3VAL到达0后，取决于T3CON[1]状态，就会产生复位或中断。为避免产生复位或者中断，向T3ICLR内写入任何值必须在T3VAL达到0之前进行。将T3LD值重新载入计数器内后就会开始一个新的超时周期。

一旦进入看门狗模式，T3LD和T3CON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。看门狗定时器应在用户代码的最初行内配置以免陷入看门狗复位的无限循环内。

定时器3在JTAG调试访问期间自动暂停，并且只在JTAG放弃ARM7内核控制权后才会重新开始计数。默认情况下，定时器3在断电期间仍会继续计数。但可通过将T3CON位0置1来禁用。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3接口

定时器3接口有4个寄存器，如表131所示。

表131. 定时器3接口寄存器

寄存器	说明
T3CON	配置寄存器。
T3LD	6位寄存器(位0至位15)；保存16位无符号整数。
T3VAL	6位寄存器(位0至位15)；保存16位无符号整数。这是一个只读寄存器。
T3CLR1	8位寄存器。正常模式下向这个寄存器写入任意值会清除定时器3中断；在看门狗模式下向这个寄存器写入任意值就会重新开始一个超时周期。

### 定时器3载入寄存器

该16位寄存器用于保存定时器3的重新载入值。

名称：	T3LD
地址：	0xFFFFF0360
默认值：	0x3BF8
访问类型：	读/写

### 定时器3值寄存器

该16位只读寄存器用于保存定时器3的当前计数值。

名称：	T3VAL
地址：	0xFFFFF0364
默认值：	0x3BF8
访问类型：	只读

### 定时器3清除寄存器

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3，以防止产生看门狗定时器复位。

名称：	T3CLR1
地址：	0xFFFFF036C
默认值：	0x0000
访问类型：	只写

### 定时器3控制寄存器

该16位寄存器用于配置定时器3的工作模式，该寄存器功能的详细说明见表132。

名称：	T3CON
地址：	0xFFFFF0368
默认值：	0x0000
访问类型：	读取、只写一次

**表132. T3CON寄存器位功能描述**

位	值	说明
15:9		这些位保留，应由用户代码写入0。
8		递增/递减计数使能。 用户代码置1，配置定时器3递增计数。 用户代码清0，配置定时器3递减计数。
7		定时器3使能。 用户代码置1，使能定时器3。 用户代码清0，禁用定时器3。
6		定时器3工作模式。 用户代码置1，定时器3采用周期模式。 用户代码清0，可将定时器3的工作模式设定为自由模式。
5		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		安全清除位。 用户置1，使能安全清除。 用户清0，禁用安全清除(默认)。
3:2	00 01 10 11	定时器3时钟(32.768 kHz)预分频器。 源时钟1分频(默认)。 保留。 保留。 保留。
1		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		PD_OFF。 用户代码置1，在利用POWCON寄存器位4使外设掉电时停止定时器3。 用户代码清0，在利用POWCON寄存器位4使外设掉电时使能定时器3。

### 安全清除位(仅用于看门狗模式)

安全清除位用于实现更高层次的保护。当它被置1时，一个特殊的数值序列就必须写入T3CLRI中来避免看门狗复位。这个特殊的数值序列是由一个8位线性反馈移位寄存器(LFSR)多项式 =  $X^8 + X^6 + X^5 + X + 1$ 产生的。

在进入看门狗模式之前必须先向T3CLRI中写入一个初始值或种子。在进入看门狗模式以后，再次写入到T3CLRI的值必须与期望值相匹配。如果匹配，当计数器被重新载入时，LFSR就会进入下一状态。如果不匹配，即使计数器没有计满，也将立即产生复位。

根据这个多项式的性质，值0x00不应该作为初始种子，因为0x00会一直迫使系统立即复位。此外，LFSR的值不能被访问，且必须在软件中产生和跟踪。

一个数值序列的示例如下所示：

1. 设定定时器3为看门狗模式之前，在T3CLRI中写入初始种子0xAA。
2. 在T3CLRI中写入0xAA，定时器3被重载。
3. 在T3CLRI中写入0x37，定时器3被重载。
4. 在T3CLRI中写入0x6E，定时器3被重载。
5. 写入0x66。0xDC是期望值，看门狗将芯片复位。

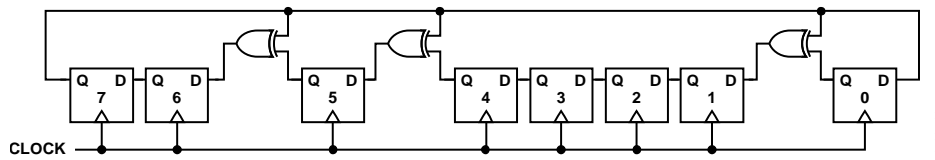


图40. 8位LFSR

### 定时器4—通用定时器

定时器4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。预分频器时钟源可以是32 kHz振荡器、内核时钟或PLL为分频输出。时钟源分频系数共有1、16、256或32,768四种。这样，当CD为0且内核在41.78 MHz频率下工作时，如预分频系数为1(忽略外部GPIO)，定时器的最小分辨率能够达到42 ns。

计数器可以采用标准的32位数格式或者时间格式(时：分：秒：百分数)。

定时器4有一个事件捕获寄存器(T4CAP)，它可以被选定的IRQ中断源初始置位所触发。中断触发时，定时器的当前值被复制到T4CAP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4接口有5个寄存器：

- T4LD、T4VAL和T4CAP都是32位寄存器，保存32位无符号整数。T4VAL和T4CAP是只读的。
- T4ICLR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。
- T4CON是配置寄存器。
- 注意，当器件处于低功耗模式且定时器4从GPIO或振荡器源接收时钟信号时，定时器4仍继续工作。

当定时器4溢出或执行T4ICLR写操作后，定时器4将从T4LD重载数值。

#### 定时器4载入寄存器

T4LD是一个32位寄存器，用于保存载入计数器的32位值。

名称： T4LD  
地址： 0xFFFF0380  
默认值： 0x00000000  
访问类型： 读/写

#### 定时器4清除寄存器

通过用户代码向该8位只写寄存器写入任意值，可以刷新(重载)定时器4。

名称： T4CLR  
地址： 0xFFFF038C  
默认值： 0x00  
访问类型： 只写

#### 定时器4值寄存器

T4VAL是一个32位寄存器，用于保存定时器4的当前值。

名称： T4VAL  
地址： 0xFFFF0384  
默认值： 0x00000000  
访问类型： 只读

#### 定时器4捕捉寄存器

这是一个32位寄存器，用于保存使能IRQ事件所捕捉的32位数值。

名称： T4CAP  
地址： 0xFFFF0390  
默认值： 0x00000000  
访问类型： 只读

#### 定时器4控制寄存器

该32位寄存器用于设置定时器4的工作模式。

名称： T4CON  
地址： 0xFFFF0388  
默认值： 0x0000  
访问类型： 读/写

# ADuC7121

**表133. T4CON寄存器位功能描述**

位	值	说明
31:18		保留。用户置0。
17		事件选择位。 用户置1, 使能对事件的实时捕捉。 用户清0, 禁用对事件的实时捕捉。
16:12		事件选择范围是0至31。事件说明参见“定时器”部分开头。
11:9		时钟选择。
	000	32.768 kHz振荡器。
	001	HCLK(内核时钟)。
	010	UCLK。
	011	UCLK。
8		递增计数。 用户置1, 定时器4递增计数。 用户清0, 定时器4递减计数(默认)。
7		定时器4使能位。 用户置1, 使能定时器4。 用户清0, 禁用定时器4(默认)。
6		定时器4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		格式。
	00	二进制(默认)。
	01	保留。
	10	时:分:秒:百分数:23小时至0小时。
	11	时:分:秒:百分数:255小时至0小时。
3:0		预分频器。
	0000	源时钟1分频(默认)。
	0100	源时钟16分频。
	1000	源时钟256分频。
	1111	源时钟32,768分频。

## 硬件设计考虑

### 电源

ADuC7121工作电压范围为3.0 V至3.6 V。模拟电源引脚和数字电源引脚(分别对应AVDD和IOVDD)是分离的,因此AVDD不受系统IOVDD线路上经常出现的干扰数字信号影响。在这种模式下,器件可以在分离电源下工作;也就是说,每个电源使用不同的电压。例如,系统的IOVDD工作电压为3.3 V,而AVDD电压为3 V;反之亦然。图41是一个典型的分离电源配置。

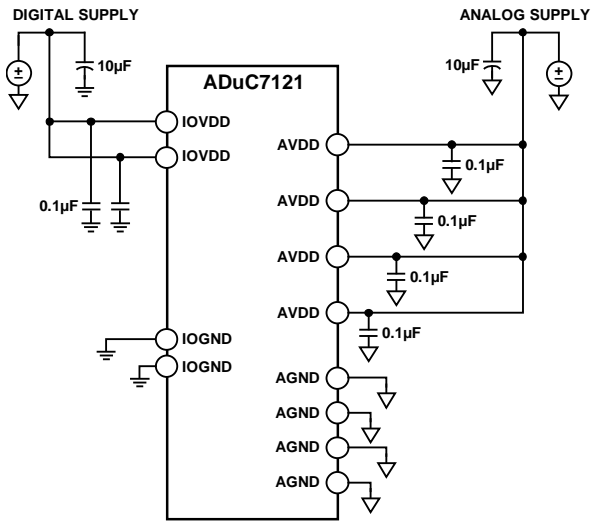


图41. 外部双电源连接

除了使用两个分离的电源外,用户还可以通过AVDD和IOVDD之间串联一个小电阻和/或磁珠来降低AVDD的噪声,然后将AVDD单独去偶到地。图42所示的就是用这种方法进行设计的一个示例。使用这种方法时,其它模拟电路(如运算放大器和基准电压源等)也可以通过AVDD供电。

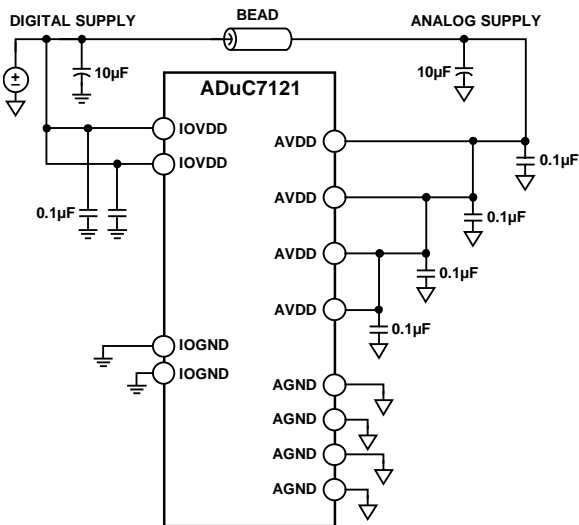


图42. 外部单电源连接

注意在图41和图42中,在IOVDD处有一个大容量值(10 μF)的储能电容,以及在AVDD处单独有一个10 μF的电容。此外,在芯片的每一个AVDD和IOVDD引脚都连接了一个小容值(0.1 μF)电容。按照实际设计时的标准,必须确保包括所有这些电容并且更小的电容应该尽可能地接近每一个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。还应注意:无论在任何时候,ADuC7121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。

### IOVDD电源灵敏度

IOVDD电源对高频噪声很敏感,因为片内振荡器和锁相环电路也是由IOVDD供电的。当内部锁相环失锁时,一个门电路会将时钟源与CPU隔离开,并且ARM7TDMI内核会停止执行代码,直到锁相环重新锁定。这个特性可以确保闪存接口时序或ARM7TDMI时序不受干扰。

通常,电源上频率高于50 kHz并且峰峰值为50 mV的噪声会导致内核停止工作。

如果在电源部分推荐的去耦电容不足以保证IOVDD上的所有噪声低于50 mV,那么就需要一个图43所推荐的滤波电路。

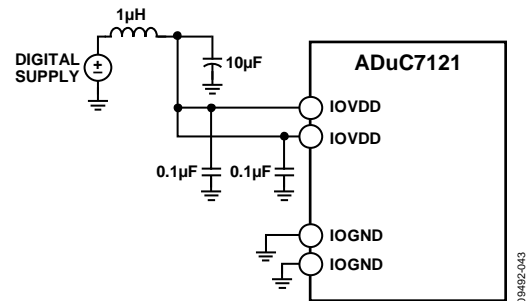


图43. 推荐的IOVDD电源滤波电路

# ADuC7121

## 线性稳压器

ADuC7121要求采用3.3 V单电源供电，而内核逻辑要求采用2.6 V电源供电。片内有一个线性稳压器，可以将来自IOVDD的电源调节到2.6 V，从而为内核逻辑供电。DVDD引脚的2.6 V电源用来给内核逻辑供电。在各DVDD和DGND之间必须连接一个0.47  $\mu\text{F}$ 的外部补偿电容(应尽量靠近这些引脚)作为电荷槽，如图44所示。内部IDAC需要2.5 V电源。内部LDO提供稳定的2.5 V电源。AVDD\_IDAC引脚的2.5 V电源用来为IDAC供电。在AVDD\_IDAC和AGND之间必须连接一个0.47  $\mu\text{F}$ 的外部补偿电容(应尽量靠近这些引脚)作为电荷槽，如图44所示。

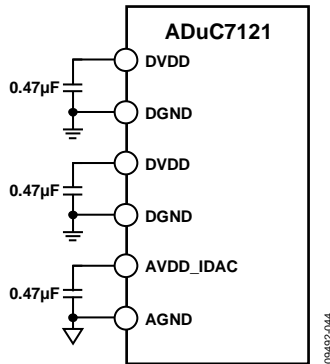


图44. 稳压器连接

DVDD引脚不能用来给任何其它芯片供电。此外，推荐在IOVDD引脚使用性能优良的电源去耦装置，以便提高片内稳压器的线性稳压性能。

## 接地和电路板布局建议

和所有高分辨率数据转换器一样，为了实现ADC和DAC的最优性能，对于基于ADuC7121的设计，应特别注意接地和PC电路板布局。

尽管这些器件已经将模拟地和数字地引脚分开(AGND和IOGND)，但用户务必不要将这些引脚连接到两个分开的地平面，除非这两个地平面非常靠近器件。图45a是一个简化的示意图。在系统中，如果数字地和模拟地平面在某处连接在一起(例如在系统电源)，那么这个平面不能再连接到靠近芯片的地方，因为这样会导致一个地环路。在这种情况下，ADuC7121的所有AGND和IOGND引脚都应连接到模拟地平面，如图45b所示。如果系统中只有一个地平面，必须确保数字和模拟器件在板上是物理分开且分属两个半平面，这样数字回路电流就不会流经模拟电路附近；反之亦然。ADuC7121可以放置在数字和模拟部分之间，如图45c所示。

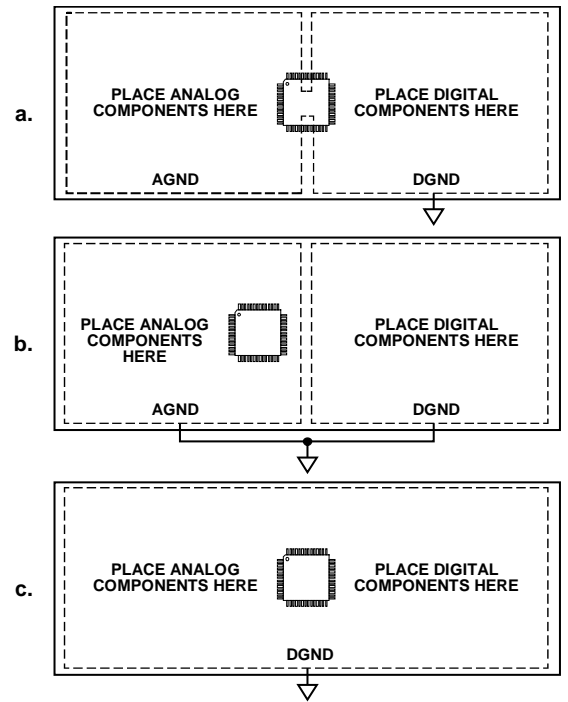


图45. 系统地框图

在上述方案以及更为复杂的实际应用中，应该特别注意来自电源的电流和返回地的电流。要确保所有电流的回流路径均尽可能靠近电流到达目的地所经过的路径。例如，不要用IOVDD给模拟部分的器件供电(如图45b所示)，因为这会导致IOVDD回路电流强行通过AGND。另外，如果电路板的左半平面放置了一个带有噪声的数字芯片(如图45c所示)，那么应该避免可能出现的数字电流流经模拟电路。如果可能的话，尽量避免在地平面上出现长的不连续部分，例如由同一层上的一条长迹线构成的不连续部分，因为它们会强迫回路信号通过一个长的路径。此外，所有需要连接到地的引脚应该直接连接到地平面，尽量少用一些或不用迹线将引脚通过其过孔与地分离。

当ADuC7121的任何数字输入引脚连接高速逻辑信号(上升/下降时间小于5 ns)时，应该在每一条相关的线路上串联一个电阻以确保器件输入引脚上信号上升和下降时间大于5 ns。通常，阻值为100  $\Omega$ 或200  $\Omega$ 的电阻足以阻止高速信号从容性器件耦合进入器件并影响ADC的转换精度。



## 时钟振荡器

ADuC7121的时钟源可以由内部锁相环或一个外部时钟输入产生。当使用内部锁相环时，应该在XTALI和XTALO引脚之间连接一个32.768 kHz的并行谐振晶体并且这两个引脚与地之间应连接一个电容，如图46所示。这个晶体使得锁相环可以正确锁相进而产生41.78 MHz频率的时钟信号。如果不使用外部晶体，内部振荡器会产生一个41.78 MHz  $\pm$  3%的典型频率。

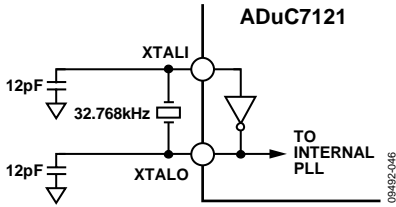


图46. 外部并行谐振晶体连接

若要使用一个外部时钟源输入来代替锁相环(如图47所示)，必须修改PLLCON寄存器的位1和位0。外部时钟从P1.4和XCLK引脚输入。

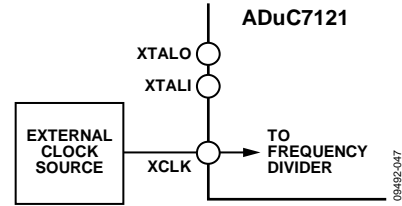
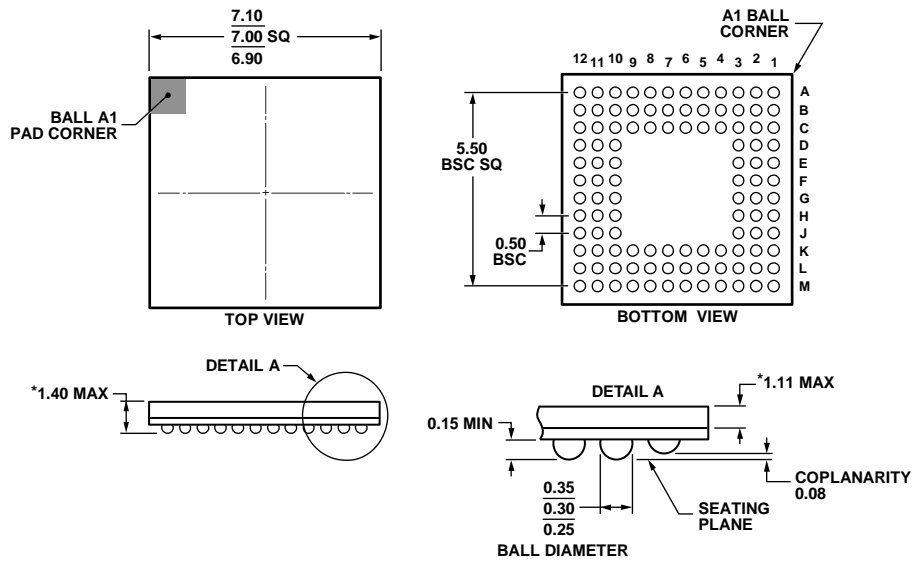


图47. 连接一个外部时钟源

使用外部时钟源时，ADuC7121的额定时钟频率范围为50 kHz到41.78 MHz  $\pm$  1%，以确保模拟外设和Flash/EE正常工作。

# ADuC7121

## 外形尺寸



\*COMPLIANT WITH JEDEC STANDARDS MO-195-BD WITH EXCEPTION TO PACKAGE HEIGHT AND THICKNESS.

图48. 108引脚CSP\_BGA封装  
(BC-108-4)

尺寸单位: mm

090408-A

### 订购指南

型号 <sup>1</sup>	温度范围	封装描述	封装选项
ADuC7121BBCZ	-10°C至+95°C	108引脚 CSP_BGA	BC-108-4
ADuC7121BBCZ-RL	-10°C至+95°C	108引脚 CSP_BGA, 13"卷带和卷盘	BC-108-4
EVAL-ADuC7121QSPZ		ADuC7121 QuickStart开发系统	

<sup>1</sup> Z = 符合RoHS标准的器件。

**注释**

## 注释

I<sup>2</sup>C指最初由Philips Semiconductors(现为NXP Semiconductors)开发的一种通信协议。