

产品特性

模拟I/O

- 13个外部通道、12位、1 MSPS ADC
- 2个具有可编程增益PGA(1至5)输入范围的差分通道
- IOVDD电源监控通道
- 片内温度传感器
- 11路通用输入
- 全差分模式和单端模式
- 模拟输入范围：0 V至 V_{REF}
- 12个12位电压输出DAC
- 片内基准电压源：1.2 V/2.5 V
- 配合外部电路使用的缓冲输出基准电压源

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I²C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BGA
- 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

- 光网络、工业控制和自动化系统
- 智能传感器和精密仪器

功能框图

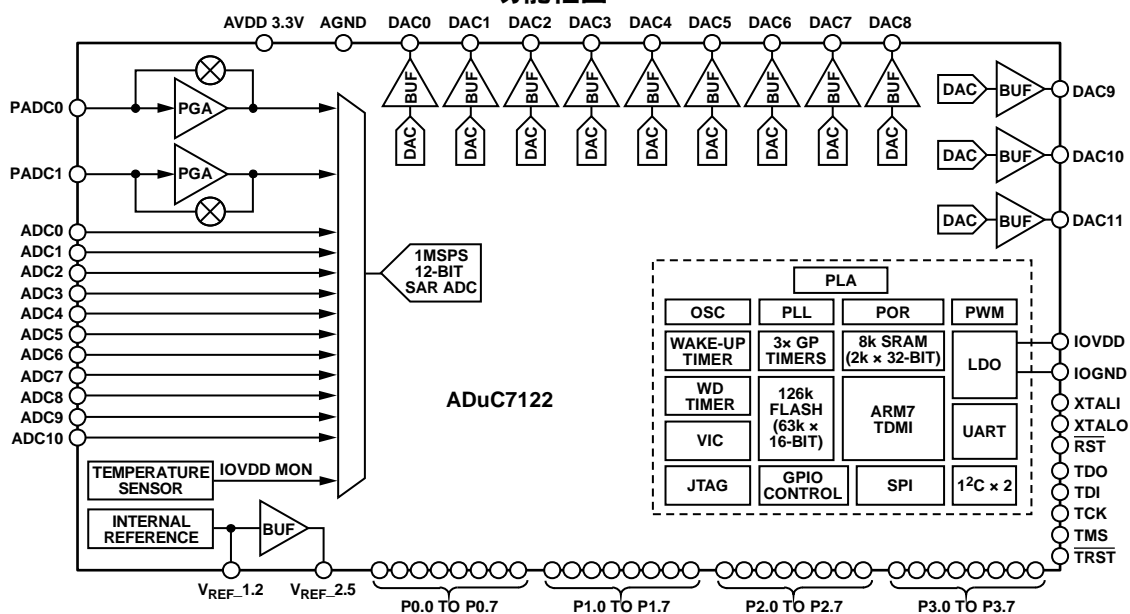


图1.

Rev. A

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

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	43
应用	1	DAC	43
功能框图	1	LDO(低压差稳压器)	45
修订历史	3	振荡器和锁相环—电源控制	46
概述	4	外部晶振选择	46
技术规格	5	外部时钟选择	46
时序规格	9	电源控制系统	47
绝对最大额定值	14	寄存器和密钥	48
ESD警告	14	数字外设	49
引脚配置和功能描述	15	PWM概述	49
术语	19	PWM转换开始控制	51
ADC规格	19	通用I/O	52
DAC技术规格	19	UART串行接口	54
ARM7TDMI内核概览	20	波特率生成	54
Thumb模式(T)	20	UART寄存器定义	54
长乘(M)	20	I ² C	59
EmbeddedICE (I)	20	串行时钟生成	59
异常	20	I ² C总线地址	59
ARM寄存器	20	I ² C寄存器	60
中断延迟时间	21	I ² C公共寄存器	68
存储器结构	22	串行外设接口	69
Flash/EE存储器	22	SPIMISO(主机输入, 从机输出)引脚	69
SRAM	22	SPIMOSI(主机输出, 从机输入)引脚	69
存储器映射寄存器	22	SPICLK(串行时钟输入/输出)引脚	69
完整MMR列表	23	SPI片选(SPICS $\overline{}$ 输入)引脚	69
ADC电路概览	27	SPI功能的外部引脚配置	69
ADC传递函数	28	SPI寄存器	70
典型操作	29	可编程逻辑阵列(PLA)	73
温度传感器	30	中断系统	76
转换器操作	33	IRQ	77
驱动模拟输入	34	快速中断请求(FIQ)	77
带隙基准电压源	34	定时器	83
电源监控器	35	时:分:秒:1/128格式	83
非易失性Flash/EE存储器	36	定时器0—使用寿命定时器	84
Flash/EE存储器概述	36	定时器1—通用定时器	85
Flash/EE存储器	36	定时器2—唤醒定时器	87
Flash/EE存储器安全性	37	定时器3—看门狗定时器	89
Flash/EE控制接口	37	定时器4—通用定时器	91
SRAM和FLASH/EE执行时间	40	硬件设计考虑	93
复位和重映射	41	电源	93

接地和电路板布局建议	94
时钟振荡器	95

外形尺寸	96
订购指南	96

修订历史

2014年11月 — 修订版0至修订版A

表1中的精度从 ± 5 mV更改为 ± 30 mV	6
更改“Flash/EE存储器”部分	22
更改“PGA和输入缓冲器”部分	28
更改“Flash/EE存储器”部分和 “串行下载(在线编程)”部分	36
更改“Flash/EE存储器安全”部分	37
更改表57和表58	40
更改I ² C部分	59
更改表101	60
更改表108	64
更改表109	65
更改表111	70
增加“时:分:秒:1/128格式”部分	83
更改表168	89
增加“硬件设计考虑”部分	93
更改“订购指南”部分	96

2010年4月 — 修订版0: 初始版

概述

ADuC7122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、12个电压输出DAC、16位/32位MCU和Flash/EE存储器。

ADC具有多达13路输入。其中4路输入可以配置为差分对，其前端的可编程增益放大器提供1至5的增益。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、温度传感器和电源电压比较器完善了ADC的外设设置。

通过编程可将DAC输出范围设置为两种电压范围之一。DAC输出具有一个增强特性，能够在看门狗或软件复位时序中保持其输出电压。

该器件通过片内振荡器和锁相环(PLL)产生41.78 MHz的内部高频时钟信号。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。

ADuC7122内置一个高级中断控制器。该矢量中断控制器(VIC)可以为每个中断分配一个优先级。它还支持嵌套中断，每个IRQ和FIQ最多允许8级嵌套。如果将IRQ和FIQ中断源合并，则可以支持总计16级嵌套中断。

片内出厂固件支持通过I²C串行接口端口进行在线下载，同时也支持通过JTAG接口进行非介入仿真。这些特性都集成在支持此MicroConverter[®]系列的低成本QuickStart™开发系统中。该器件内置一个提供6路输出信号的16位PWM。

为便于通信，该器件内置2个I²C通道，可以将这些通道独立配置为主模式或从模式。另外还提供了支持主从两种模式的SPI接口。

ADuC7122工作电压为3.0 V至3.6 V，额定温度范围为-10°C至+95°C，采用108引脚BGA封装。

技术规格

除非另有说明， $AV_{DD} = IOV_{DD} = 3.0\text{ V}$ 至 3.6 V ， $V_{REF} = 2.5\text{ V}$ 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		μs	
直流精度 ^{1,2}	12			位	
分辨率		± 0.6	± 2	LSB	2.5 V内部基准电压，PADC0/PADC1通道未经生产测试
积分非线性		± 0.5	$+1.4/-0.99$	LSB	2.5 V内部基准电压，保证单调性
微分非线性 ^{3,4}		1		LSB	ADC输入为直流电压
直流代码分布					
端点误差 ⁵					内部无缓冲通道
失调误差		± 2	± 5	LSB	
失调误差匹配		± 1		LSB	
增益误差		± 2	± 5	LSB	
增益误差匹配		± 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	参见表35和表36
单端模式			0至 V_{REF}	V	缓冲旁路
单端模式	0.15		$AV_{DD} - 1.5$	V	缓冲使能
漏电流		± 0.2		μA	
输入电容		20		pF	ADC采样期间，缓冲旁路
输入电容		20		pF	ADC采样期间，缓冲使能
PADC0输入					28.3 k Ω 电阻，PGA增益 = 3，采集时间 = 6 μs ，伪差分模式
满量程输入范围	20		1000	μA	
PADC0P的输入漏电流 ⁴		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 ⁴			1	%	
增益漂移 ⁴			50	ppm/ $^\circ\text{C}$	
失调 ⁴		3	6	nA	不包括PGA失调
失调漂移 ⁴		30	60	pA/ $^\circ\text{C}$	
PADC0P顺从电压范围	0.1		$AV_{DD} - 1.2$	V	
PADC1输入					53.5 k Ω 电阻，PGA增益 = 3，采集时间 = 6 μs ，伪差分模式
满量程输入范围	10.6		700	μA	
PADC1P的输入漏电流 ⁴		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 ⁴			1	%	
增益漂移 ⁴			50	ppm/ $^\circ\text{C}$	
失调 ⁴		3	6	nA	不包括PGA失调
失调漂移 ⁴		30	60	pA/ $^\circ\text{C}$	
PADC1P顺从电压范围	0.1		$AV_{DD} - 1.2$	V	

ADuC7122

参数	最小值	典型值	最大值	单位	测试条件/注释
片内基准电压 输出电压 精度 ⁷ 基准电压温度系数 ⁴ 电源抑制比 输出阻抗 内部V _{REF} 上电时间		2.5		V mV ppm/°C dB Ω ms	在V _{REF} 和AGND之间连接0.47 μF电容 T _A = 25°C T _A = 25°C
外部基准输入 输入电压范围	1.2		AV _{DD}	V	
BUF_VREF1、BUF_VREF2输出 精度 基准源温度系数 负载电流		±30 40		mV μV/°C mA	T _A = 25°C
DAC通道规格 直流精度 ⁸ 分辨率 相对精度 微分非线性 计算失调误差 实际失调误差 增益误差 ⁹ 增益误差失配 建立时间				位 LSB LSB mV mV % % μs	R _L = 5 kΩ, C _L = 100 pF 缓冲 保证单调性 2.5 V内部基准电压 在代码0下测量 DAC0满量程的百分比
PSRR ⁴ 直流 1 kHz 10 kHz 100 kHz				dB dB dB dB	缓冲
失调漂移 ⁴			10	μV/°C	
增益误差漂移 ⁴			10	μV/°C	
短路电流		20		mA	
模拟输出 输出范围	0.1		V _{REF} /AV _{DD} - 0.1	V	缓冲开启
DAC交流特性 压摆率 输出电压建立时间 数模转换脉冲干扰		2.49 10 ±20		V/μs μs nV-sec	主进位1 LSB变化(DACxDAT寄存器中同时变化的最大位数)
温度传感器 ¹⁰ 25°C时电压输出 电压TC 精度		707 -1.25 ±3		mV mV/°C °C	测量前MCU处于掉电或待机模式
电源监控器(PSM) IOV _{DD} 跳变点选择 电源跳变点精度		2.79 3.07 ±2.5		V V %	两个可选择跳变点 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	

参数	最小值	典型值	最大值	单位	测试条件/注释
FLASH/EE存储器 耐久性 ¹¹ 数据保留期 ¹²	10,000 20			周期 年	$T_J = 85^\circ\text{C}$
数字输入 逻辑1输入电流 逻辑0输入电流 输入电容		± 0.2 -40 10	± 1 -60	μA μA pF	除XTALI和XTALO外的所有数字输入 $V_{IH} = V_{DD}$ 或 $V_{IH} = 5\text{V}$ $V_{IL} = 0\text{V}$; TDI除外
逻辑输入 ⁴ 输入低电压 V_{INL}^4 输入高电压 V_{INH}^4	2.0		0.8	V V	除XTALI外的所有逻辑输入
逻辑输出 输出高电压 V_{OH} V_{OL} 输出低电压 ¹³	2.4		0.4	V V	除XTALO外的所有数字输出 $I_{SOURCE} = 1.6\text{mA}$ $I_{SINK} = 1.6\text{mA}$
晶体输入XTALI和XTALO 逻辑输入, 仅限XTALI 输入低电压 V_{INL} 输入高电压 V_{INH} XTALI输入电容 XTALO输出电容		1.1 1.7 20 20		V V pF pF	
内部振荡器		32.768	± 3	kHz %	
MCU时钟速率 采用32 kHz内部振荡器 采用32 kHz外部晶体 使用外部时钟		326 41.78	41.78	kHz MHz MHz	CD = 7 CD = 0 $T_A = 95^\circ\text{C}$
启动时间 上电时 从暂停/休眠模式 从休眠模式 从停止模式		70 24 3.06 1.58 1.7		ms ns μs ms ms	内核时钟= 41.78 MHz CD = 0 CD = 7
可编程逻辑阵列(PLA) 引脚传输延迟 单元传输延迟		12 2.5		ns ns	从输入引脚到输出引脚
电源要求 ^{14, 15} 电源电压范围 AV_{DD} 至AGND和 IOV_{DD} 至IOGND 模拟电源电流 AV_{DD} 电流 数字电源电流 IOV_{DD} 正常模式下电流 IOV_{DD} 暂停模式下电流 ⁴ IOV_{DD} 休眠模式下电流 ⁴ 附加电源电流 模数转换器 DAC	3.0	200 7 11 30 25 100 2.7 250	3.6 40	V μA mA mA mA mA μA mA μA	ADC处于空闲模式 从Flash/EE执行代码 CD = 7 CD = 3 CD = 0(时钟频率41.78 MHz) CD = 0(时钟频率41.78 MHz) $T_A = 85^\circ\text{C}$ 1 MSPS时 每DAC

ADuC7122

参数	最小值	典型值	最大值	单位	测试条件/注释
ESD测试					2.5 V基准电压, $T_A = 25^\circ\text{C}$
最大HBM通过电压			4	kV	
最大FCIDM通过电压			0.5	kV	

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

² 适用于所有ADC输入通道。

³ 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; 参见“校准”部分。

⁴ 未经生产测试, 但量产时的设计和/或特性数据可提供保证。

⁵ 采用运算放大器AD845作为一个外部输入缓冲级, 用ADCOF和ADCGN寄存器中的出厂设定默认值进行测试(如图23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(详见“ADC电路概览”部分)。

⁶ 输入信号可以任何直流共模电压(V_{CM})为中心, 但该值必须位于ADC规定输入电压范围内。

⁷ V_{REF} 校准和调整是在以下条件下进行: 内核工作在正常模式($CD = 0$)、ADC开启且所有DAC开启。在其他工作条件下, V_{REF} 精度可能有所差异。

⁸ 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。

⁹ 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。

¹⁰ 芯片温度。

¹¹ 耐久性是在分别在 -40°C 、 $+25^\circ\text{C}$ 、 $+85^\circ\text{C}$ 及 $+125^\circ\text{C}$ 时依据JEDEC 22标准方法A117来认定的。

¹² 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。

¹³ 测试是在最多8个I/O端口输出低电平时进行的。

¹⁴ 电源功耗分别在正常、暂停和休眠模式下测试的, 这3种模式下的测试条件分别为: 正常模式供电电压为3.6V、暂停模式供电电压为3.6V、休眠模式供电电压为3.6V。

¹⁵ 在一个Flash/EE擦写周期中, IOV_{DD} 电源电流通常降低2 mA。

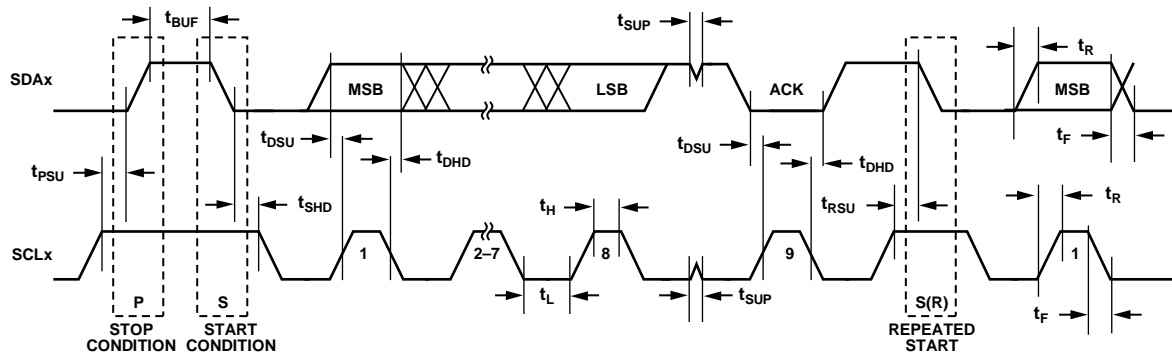
时序规格

表2. 快速模式下I²C时序(400 kHz)

参数	说明	从机			主机			单位
		最小值	典型值	最大值	最小值	典型值	最大值	
t_L	SCLx低电平脉冲宽度	200			1360			ns
t_H	SCLx高电平脉冲宽度	100			1140			ns
t_{SHD}	起始条件保持时间	300						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						ns
t_{PSU}	停止条件的建立时间	100			800			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						μ s
t_R	SCLx和SDAx的上升时间			300			200	ns
t_F	SCLx和SDAx的下降时间			300				ns

表3. 标准模式下I²C时序(100 kHz)

参数	说明	从机			单位
		最小值	典型值	最大值	
t_L	SCLx低电平脉冲宽度	4.7			μ 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			μ s
t_{DSU}	数据建立时间	250			ns
t_{DHD}	数据保持时间	0		3.45	μ s
t_{RSU}	重复起始建立时间	4.7			μ s
t_{PSU}	停止条件的建立时间	4.0			μ 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			μ s
t_R	SCLx和SDAx的上升时间			1	μ s
t_F	SCLx和SDAx的下降时间			300	ns

图2. I²C兼容接口时序

ADuC7122

表4. SPI主机模式时序(SPICPH = 1)

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

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

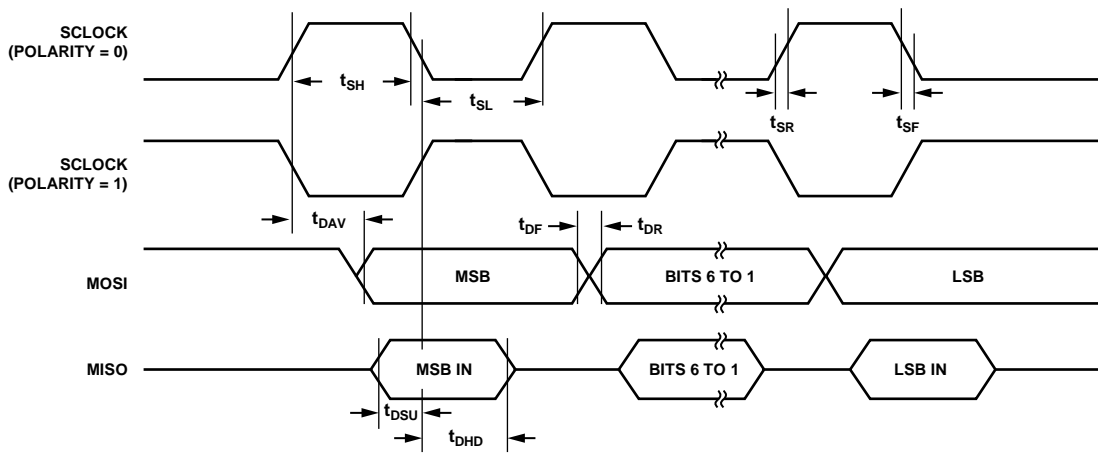


图3. SPI主机模式时序(SPICPH = 1)

08755-03

ADuC7122

表6. SPI从机模式时序(SPICPH = 1)

参数	说明	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	\overline{CS} 至SCLOCK边沿	200			ns
t_{SL}	SCLOCK低电平脉冲宽度 ¹		$(SPIDIV + 1) \times t_{uCLK}$		ns
t_{SH}	SCLOCK高电平脉冲宽度 ¹		$(SPIDIV + 1) \times t_{uCLK}$		ns
t_{DAV}	SCLOCK边沿之后数据输出有效			25	ns
t_{DSU}	SCLOCK边沿之前数据输入建立时间	$1 \times t_{uCLK}$			ns
t_{DHD}	SCLOCK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}	SCLOCK上升时间		5	12.5	ns
t_{SF}	SCLOCK下降时间		5	12.5	ns
t_{SFS}	SCLOCK边沿之后 \overline{CS} 高电平时间	0			ns

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

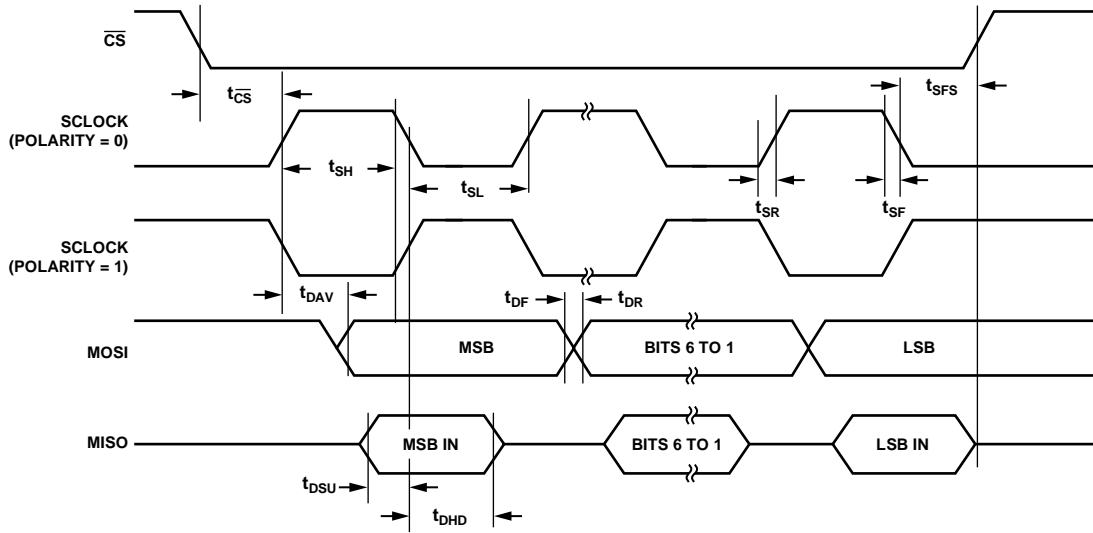


图5. SPI从机模式时序(SPICPH = 1)

087595-005

表7. SPI从机模式时序(SPICPH = 0)

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

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

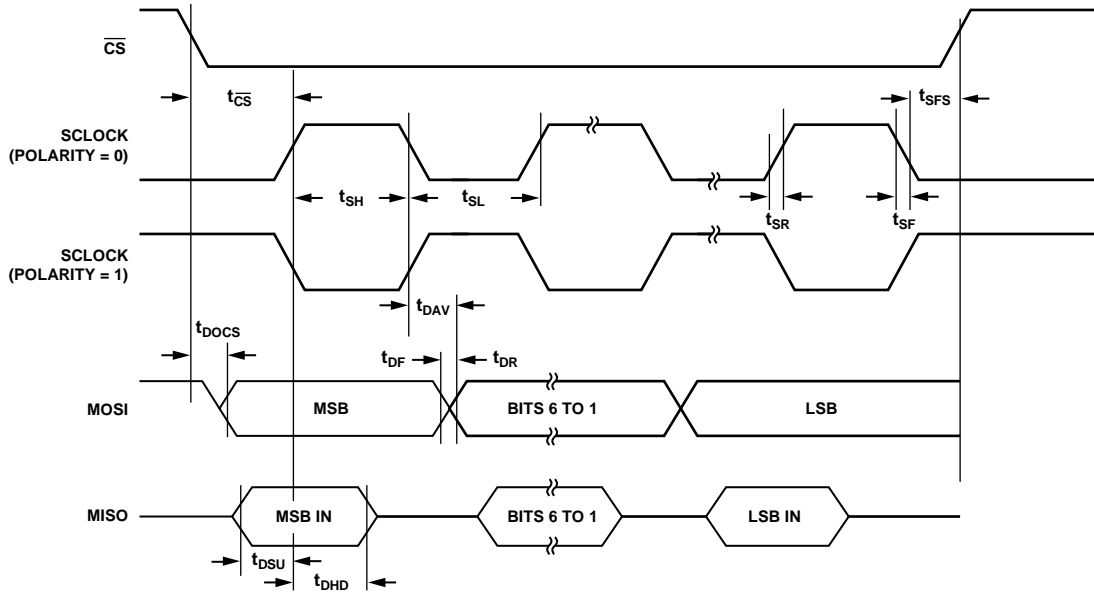


图6. SPI从机模式时序(SPICPH = 0)

087755-006

绝对最大额定值

除非另有说明，AGND = REFGND = DACGND = GND_{REF}，

T_A = 25°C。

表8.

参数	额定值
AV _{DD} 至IOV _{DD}	-0.3V至+0.3V
AGND至DGND	-0.3V至+0.3V
IOVDD至IOGND，AVDD至AGND	-0.3V至+6V
数字输入电压至IOGND	-0.3V至+5.3V
数字输出电压至IOGND	-0.3V至IOV _{DD} + 0.3V
V _{REF-2.5} 和V _{REF-1.2} 至AGND	-0.3V至AV _{DD} + 0.3V
模拟输入至AGND	-0.3V至AV _{DD} + 0.3V
模拟输出至AGND	-0.3V至AV _{DD} + 0.3V
工作温度范围	-10°C至+95°C
存储温度范围	-65°C至+150°C
结温	150°C
θ _{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

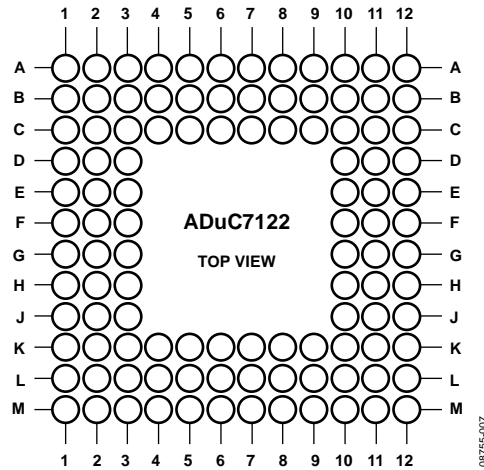


图7. 引脚配置

表9. 引脚功能描述

引脚编号	引脚名称	类型 ¹	说明
C12	RST	I	复位输入(低电平有效)。
D11	P0.0/SCL1/PLAI[5]	I/O	通用输入和输出端口0.0 (P0.0)。 I2C0的I ² C接口SCLOCK (SCL1)。 PLA单元5输入(PLAI[5])。
E11	P0.1/SDA1/PLAI[4]	I/O	通用输入和输出端口0.1 (P0.1)。 I2C0的I ² C接口SDATA (SDA1)。 PLA单元4输入(PLAI[4])。
C3	P0.2/SPICLK/ADC _{Busy} /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 _{Busy})。 PLA单元13输出(PLAO[13])。
D3	P0.3/SPIMISO/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(SPIMISO)。 PLA单元12输出(PLAO[12])。
E3	P0.4/SPIMOSI/PLAI[11]/TRIP	I/O	利用外部源同步复位PWM计数器的输入(SYNC)。 通用输入和输出端口0.4 (P0.4)。 SPI主机输出、从机输入(SPIMOSI)。 PLA单元11输入(PLAI[11])。
F3	P0.5/SPICS/PLAI[10]/CONVST	I/O	允许触发PWM中断的输入(TRIP)。 通用输入和输出端口0.5 (P0.5)。 SPI从机选择输入(SPICS)。 PLA单元10输入(PLAI[10])。 利用PLA或定时器输出启动ADC转换(CONVST)。
G3	P0.6/MRST/PLAI[2]	I/O	通用输入和输出端口0.6 (P0.6)。 上电复位输出(MRST)。 PLA单元2输入(PLAI[2])。
G10	P0.7/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(TRST)。调试和下载访问。 PLA单元3输入(PLAI[3])。
C2	P1.0/SIN/SCL2/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(RxD)，UART (SIN)。 I2C1的I ² C接口SCLOCK (SCL2)。 PLA单元7输入(PLAI[7])。
D2	P1.1/SOUT/SDA2/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(TxD)，UART (SOUT)。 I2C1的I ² C接口SDATA (SDA2)。 PLA单元6输入(PLAI[6])。

ADuC7122

引脚编号	引脚名称	类型 ¹	说明
H3	P1.4/PWM1/PLAI[8]/ECLK/XCLK	I/O	通用输入和输出端口1.4 (P1.4)。PWM1输出(PWM1)。PLA单元8输入(PLAI[8])。基本系统时钟输出(ECLK)。基本系统时钟输入(XCLK)。
J3	P1.5/PWM2/PLAI[9]	I/O	通用输入和输出端口1.5 (P1.5)。PWM2输出(PWM2)。PLA单元9输入(PLAI[9])。
B3	P1.6/PLAO[5]	I/O	通用输入和输出端口1.6 (P1.6)。PLA单元5输出(PLAO[5])。
B2	P1.7/PLAO[4]	I/O	通用输入和输出端口1.7 (P1.7)。PLA单元4输出(PLAO[4])。
F11	P2.0/IRQ0/PLAI[13]	I/O	通用输入和输出端口2.0 (P2.0)。外部中断请求0 (IRQ0)。PLA单元13输入(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 (IRQ1)。PLA单元12输入(PLAI[12])。
H11	P2.2/PLAI[1]	I/O	通用输入和输出端口2.2 (P2.2)。PLA单元1输入(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 (IRQ2)。PLA单元14输入(PLAI[14])。
H10	P2.4/PWM5/PLAO[7]	I/O	通用输入和输出端口2.4 (P2.4)。PWM5输出(PWM5)。PLA单元7输出(PLAO[7])。
J10	P2.5/PWM6/PLAO[6]	I/O	通用输入和输出端口2.5 (P2.5)。PWM6输出(PWM6)。PLA单元6输出(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 (IRQ3)。PLA单元15输入(PLAI[15])。
C9	P2.7/PLAI[0]	I/O	通用输入和输出端口2.7 (P2.7)。PLA单元0输入(PLAI[0])。
C4	P3.0/PLAO[0]	I/O	通用输入和输出端口3.0 (P3.0)。PLA单元0输出(PLAO[0])。
C11	P3.1/PLAO[1]	I/O	通用输入和输出端口3.1 (P3.1)。PLA单元1输出(PLAO[1])。
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 (IRQ4)。PWM3输出(PWM3)。PLA单元2输出(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 (IRQ5)。PWM4输出(PWM4)。PLA单元3输出(PLAO[3])。
E2	P3.4/PLAO[8]	I/O	通用输入和输出端口3.4 (P3.4)。PLA单元8输出(PLAO[8])。
F2	P3.5/PLAO[9]	I/O	通用输入和输出端口3.5 (P3.5)。PLA单元9输出(PLAO[9])。
D12	P3.6/PLAO[10]	I/O	通用输入和输出端口3.6 (P3.6)。PLA单元10输出(PLAO[10])。

引脚编号	引脚名称	类型 ¹	说明
E12	P3.7/BM/PLAO[11]	I/O	通用输入和输出端口3.7 (P3.7)。引导模式(BM)。如果BM为低电平且闪存地址0x00014为0xFFFFFFFF，则在下一个复位序列之后，器件进入I ² C下载模式。PLA单元11输出(PLAO[11])。
L8	V _{REF_2.5}	AI/O	2.5 V基准电压输出，外部2.5 V基准电压输入。可用于驱动光电二极管的阳极。
L5	V _{REF_1.2}	AI/O	1.2 V基准电压输出，外部1.2 V基准电压输入。不能用于向外部流出电流。
B8	NC	NC	不连接。
K6	BUF_VREF1	AO	缓冲2.5 V偏置。最大负载 = 1.2 mA。
K7	BUF_VREF2	AO	缓冲2.5 V偏置。最大负载 = 1.2 mA。
L6	PADC0P	AI	PADC0正输入通道。基于PGA的ADC输入通道。
M5	PADC0N	AI	PADC0负输入通道。基于PGA的ADC输入通道。
L7	PADC1P	AI	PADC1正输入通道。基于PGA的ADC输入通道。
M8	PADC1N	AI	PADC1负输入通道。基于PGA的ADC输入通道。
K5	ADC0	AI	单端或差分模拟输入0。
K4	ADC1	AI	单端或差分模拟输入1。
M4	ADC2	AI	单端或差分模拟输入2。
L4	ADC3	AI	单端或差分模拟输入3。
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。伪差分输入共模(AINCM)。
K1	DAC0	AO	12位DAC输出。
K2	DAC1	AO	12位DAC输出。
J2	DAC2	AO	12位DAC输出。
L2	DAC3	AO	12位DAC输出。
M2	DAC4	AO	12位DAC输出。
L3	DAC5	AO	12位DAC输出。
M11	DAC6	AO	12位DAC输出。
L11	DAC7	AO	12位DAC输出。
L10	DAC8	AO	12位DAC输出。
K10	DAC9	AO	12位DAC输出。
K11	DAC10	AO	12位DAC输出。
K12	DAC11	AO	12位DAC输出。
B5	NC	NC	不连接。
C6	NC	NC	不连接。
A6	NC	NC	不连接。
A8	NC	NC	不连接。
A7	NC	NC	不连接。
C8	NC	NC	不连接。
A5	NC	NC	不连接。
C5	NC	NC	不连接。
B4	NC	NC	不连接。
A4	NC	NC	不连接。
A1	NC	NC	不连接。
A3	NC	NC	不连接。
A2	NC	NC	不连接。
B1	NC	NC	不连接。
A12	NC	NC	不连接。

ADuC7122

引脚编号	引脚名称	类型 ¹	说明
A9	NC	NC	不连接。
A11	NC	NC	不连接。
A10	NC	NC	不连接。
B12	NC	NC	不连接。
B11	NC	NC	不连接。
B10	AGND	S	模拟地。
B9	AGND	S	模拟地。
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	NC	NC	模拟电源(3.3 V)。
B7	REG_PWR	S	2.5 V片内稳压器的输出。必须在此引脚与DGND之间连接一个470 nF电容。
G1	LVDD	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	LVDD	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	晶体振荡反相器输出。如果不使用外部晶体，此引脚可以保持断开状态。
H2	XTALI	DI	晶体振荡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。
D10	TDO/P1.3/PLAO[14]	DO	JTAG测试端口输出，测试数据输出(TDO)。调试和下载访问。 通用输入和输出端口1.3 (P1.3)。 PLA单元14输出(PLAO[14])。通过JTAG接口调试时，此引脚不应用作GPIO。
C10	TDI/P1.2/PLAO[15]	DI	JTAG测试端口输入，测试数据输入(TDI)。调试和下载访问。 通用输入和输出端口1.2 (P1.2)。 PLA单元15输出(PLAO[15])。通过JTAG接口调试时，此引脚不应用作GPIO。
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。

¹ I = 输入，I/O = 输入/输出，AI/O = 模拟输入/输出，NC = 不连接，AO = 模拟输出，AI = 模拟输入，DI = 数字输入，DO = 数字输出，S = 电源。

术语

ADC规格

积分非线性(INL)

ADC输出与通过ADC端点的传递函数直线之间的最大偏差。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(DNL)

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

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

增益误差

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(SINAD)

在ADC输出端所测量到的信号与(噪声+失真)的比值。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，信纳比值理论值计算公式为：

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

因此，对于12位转换器，该值为74 dB。

总谐波失真

所有谐波均方根和与基波均方根之比。

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

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是ARM7内核，还有4个额外的特性：

- 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，处理器会暂停运行，进入调试状态。一旦进入调试状态，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。所有的异常模式都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。

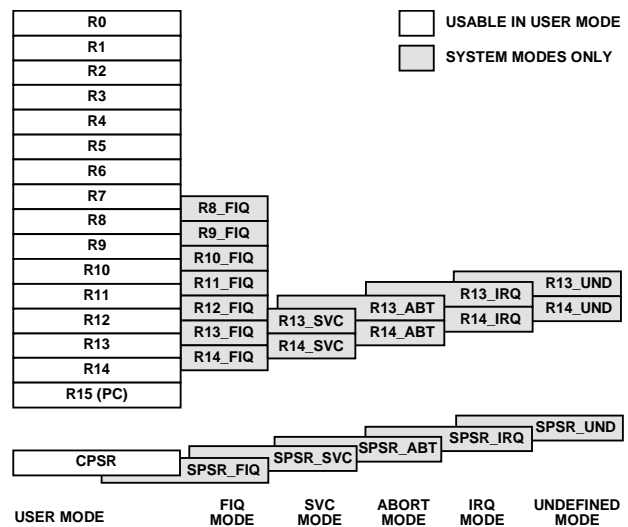


图 8. 寄存器结构图

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：

- DDI 0029G, ARM7TDMI技术参考手册
- 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，略微小于1.2 μ s。

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

ADuC7122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。

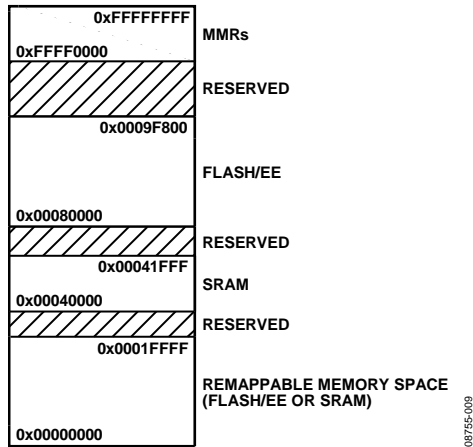


图9. 物理存储器映射图

存储器访问

ARM7内核把存储器看成是 2^{32} 个字节的线性阵列。不同的存储器模块映射如图9所示。

ADuC7122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。

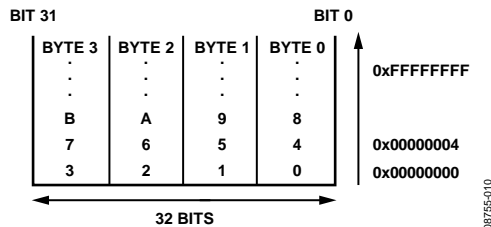


图10. 从小到大顺序格式

FLASH/EE存储器

128 kB的Flash/EE分为两个32k 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，31k x 16位是用户空间，1k 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32k x 16位，整个模块均用作用户空间。

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41.78 MHz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26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，ADuC7122中的所有外围设备均位于APB上。

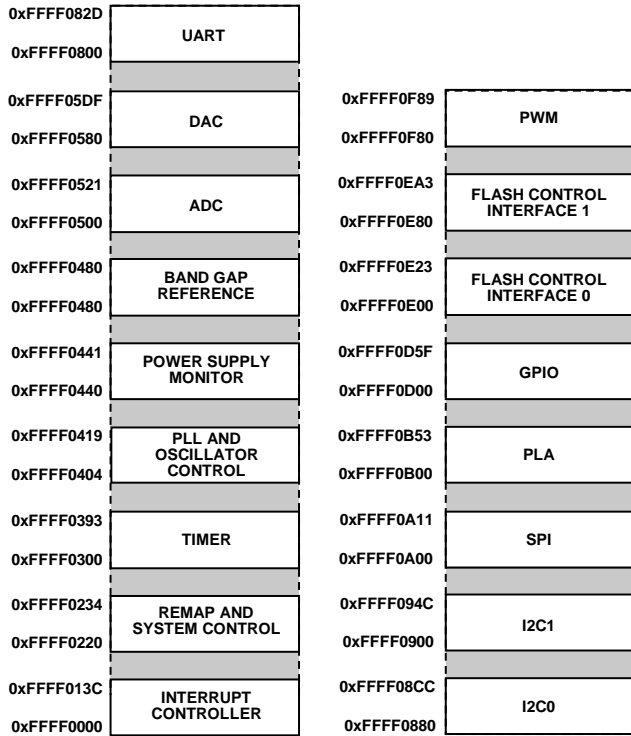


图 11. 存储器映射寄存器

完整MMR列表

注意：表10至表26中的“访问类型”栏对应于读取或写入寄存器的访问时间。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	W	1
0x0010	SWICFG	4	W	1
0x0014	IRQBASE	4	R/W	1
0x001C	IRQVEC	4	R	1
0x0020	IRQP0	4	R/W	1
0x0024	IRQP1	4	R/W	1
0x0028	IRQP2	4	R/W	1
0x002C	IRQP3	4	R/W	1
0x0030	IRQCONN	1	R/W	1
0x0034	IRQCONE	1	R/W	1
0x0038	IRQCLRE	1	W	1
0x003C	IRQSTAN	1	R/W	
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	RSTKEY1	1	W	N/A
0x024C	RSTCFG	1	R/W	0x00
0x0250	RSTKEY2	1	W	N/A

ADuC7122

表12. 定时器基地址 = 0xFFFF0300

地址	名称	字节	访问类型	周期
0x0300	TOLD	2	R/W	2
0x0304	T0VAL0	2	R	2
0x0308	T0VAL1	4	R	2
0x030C	T0CON	4	R/W	2
0x0310	T0CLRI	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	T1CLRI	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	T2CLRI	1	W	2
0x0360	T3LD	2	R/W	2
0x0364	T3VAL	2	R	2
0x0368	T3CON	2	R/W	2
0x036C	T3CLRI	1	W	2
0x0380	T4LD	4	R/W	2
0x0384	T4VAL	4	R	2
0x0388	T4CON	4	R/W	2
0x038C	T4CLRI	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
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
0x0590	DAC2CON	2	R/W	2
0x0594	DAC2DAT	4	R/W	2
0x0598	DAC3CON	2	R/W	2
0x059C	DAC3DAT	4	R/W	2
0x05A0	DAC4CON	2	R/W	2
0x05A4	DAC4DAT	4	R/W	2
0x05A8	DAC5CON	2	R/W	2
0x05AC	DAC5DAT	4	R/W	2
0x05B0	DAC6CON	2	R/W	2
0x05B4	DAC6DAT	4	R/W	2
0x05B8	DAC7CON	2	R/W	2
0x05BC	DAC7DAT	4	R/W	2
0x05C0	DAC8CON	2	R/W	2
0x05C4	DAC8DAT	4	R/W	2
0x05C8	DAC9CON	2	R/W	2
0x05CC	DAC9DAT	4	R/W	2
0x05D0	DAC10CON	2	R/W	2
0x05D4	DAC10DAT	4	R/W	2
0x05D8	DAC11CON	2	R/W	2
0x05DC	DAC11DAT	4	R/W	2

表18. UART0基地址 = 0xFFFF0800

地址	名称	字节	访问类型	周期
0x0800	COMTX	1	W	2
0x0800	COMRX	1	R	2
0x0800	COMDIV0	1	R/W	2
0x0804	COMIEN0	1	R/W	2
0x0804	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
0x0818	COMSTA1	1	R	2
0x081C	COMSCR	1	R/W	2
0x0820	COMIEN1	1	R/W	2
0x0824	COMIID1	1	R	2
0x0828	COMADR	1	R/W	2
0x082C	COMDIV2	2	R/W	2

表19. I2C0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	R/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	R/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20. I2C1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	R/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	R/W	2
0x0938	I2C1ALT	1	R/W	2
0x093C	I2C1ID0	1	R/W	2
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21. SPI基地址 = 0xFFFF0A00

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

ADuC7122

表22. 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	PLACK	1	R/W	2
0x0B44	PLAIRQ	4	R/W	2
0x0B48	PLAADC	4	R/W	2
0x0B4C	PLADIN	4	R/W	2
0x0B50	PLADOUT	4	R	2

表23. GPIO基地址 = 0xFFFF0D00

地址	名称	字节	访问类型	周期
0x0D00	GP0CON	4	R/W	1
0x0D04	GP1CON	4	R/W	1
0x0D08	GP2CON	4	R/W	1
0x0D0C	GP3CON	4	R/W	1
0x0D20	GP0DAT	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
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
0x0D70	GP1OCE	1	W	1
0x0D74	GP2OCE	1	W	1
0x0D78	GP3OCE	1	W	1

表24. 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25. 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26. 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至 V_{REF} 。在全差分模式下, 输入信号共模电压 V_{CM} 在0至 AV_{DD} 之间, 最大幅值不超过 $2V_{REF}$ (见图12)。

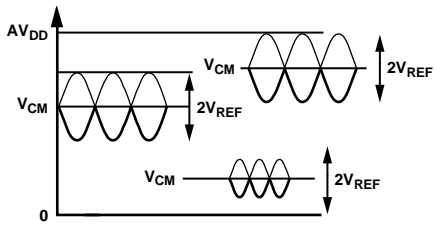


图12. 全差分模式下的平衡信号示例

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 \overline{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(相当于一个附加ADC通道输入)路由, 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 $\pm 3^{\circ}\text{C}$ 。

ADuC7122对ADuC702x系列的ADC输入结构做了多处修改。

PADC0和PADC1输入在伪差分模式下连接到PGA, 提供从1到5的32步可选增益。其余ADC通道可配置为单端、差分或伪差分模式。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

在伪差分或单端模式中，输入电压范围为0 V至 V_{REF} ，输出编码为标准二进制编码：

$$1 \text{ LSB} = FS/4096 \text{ 或}$$

$$2.5 \text{ V}/4096 = 0.61 \text{ mV} \text{ 或}$$

$$610 \mu\text{V} \text{ (} V_{REF} = 2.5 \text{ V)}$$

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。

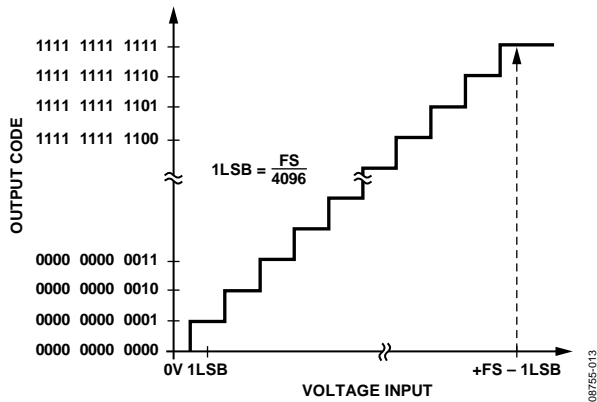


图13. 伪差分或单端模式下ADC传递函数

全差分模式

差分信号的幅值为当前使能差分通道的输入 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 $+V_{REF}$ 峰值(即 $2 \times V_{REF}$)。这与共模电压(CM)无关。共模电压是两信号的平均值 $(V_{IN+} + V_{IN-})/2$ ，同时也是两个输入电压的中点。这使得各输入的范围为 $CM \pm V_{REF}/2$ 。这一输入电压必须在外部设定，并且它的范围随着 V_{REF} 而变化(参见“驱动模拟输入”部分)。

当 $V_{REF} = 2.5 \text{ V}$ 时，全差分模式下，输出编码为二进制补码，且 $1 \text{ LSB} = 2 V_{REF}/4096$ ，或 $2 \times 2.5 \text{ V}/4096 = 1.22 \text{ mV}$ 。输出结果为 ± 11 位，但是向右移了一位，这样在书写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。

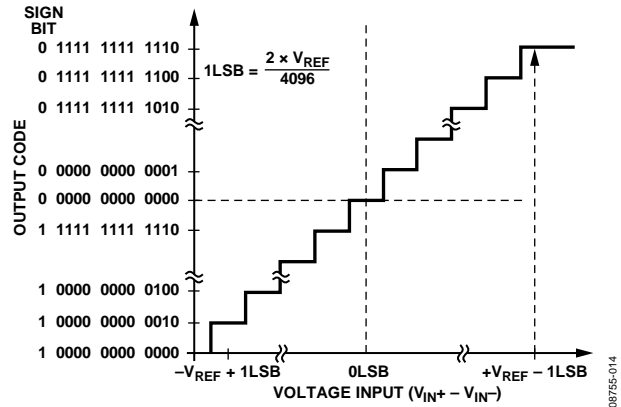


图14. 差分模式下ADC传递函数

ADC输入通道

ADuC7122提供11个固定增益ADC输入引脚。每个引脚都可以独立配置为差分输入对、单端输入或正端伪差分输入(负端必须是AINCM通道)。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 $AV_{DD} - 0.15 \text{ V}$ 。如果输入信号超过此范围，则必须旁路输入缓冲器。

可配置ADC多路复用器来选择IOVDD_MON等内部通道或温度传感器。在内部通道上转换时，必须使能输入缓冲器。

此外，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

ADuC7122内置两个可编程增益通道，它们以伪差分模式工作。PGA是一个单级正增益放大器，支持的输入范围是 0.1 V 至 $AV_{DD} - 1.2 \text{ V}$ ，输出摆幅可达 2.5 V 。PGA设计用于处理最小 10 mV 输入。

PGA的增益是从1到5，共有32个线性步进。PADC0和PADC1通道的PGA不能旁路。

PGA采用PMOS输入以使非线性度和噪声最小。为确保放大器不发生饱和，PGA的输入电平应介于 $AV_{DD} - 1.2 \text{ V}$ 和 0.1 V 之间。输入缓冲器为轨到轨缓冲器，支持 0.15 V 至 $AV_{DD} - 0.15 \text{ V}$ 的信号。每个输入缓冲器都可以独立地旁路。

为使噪声最低，可以旁路PADC输入缓冲器。

当输入电流降为0时，PADCxN由一个缓冲器驱动到 0.15 V ，以防PGA饱和。也可以将ADCCON[14]置1，以禁用该缓冲器，使得PADCxN可以连接到GND。

PADCx通道只能在伪差分模式下工作，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，用户必须执行以下步骤：

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. 设置PGA的适当增益值。
4. 将ADC设置为伪差分模式并启动转换。

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ADCDAT寄存器中提供12位结果。

高4位是符号位，12位结果放在位16至位27，如图15所示。注意：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。

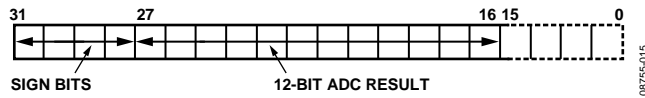


图15. ADC转换结果格式

校准

ADC失调寄存器(ADCOF)和增益系数寄存器为出厂设定值时，单次操作的端点误差和线性度可以达到最佳效果(参见“概述”部分)。如果需要进行系统校准，可以通过修改失调和增益系数的默认值来改善端点误差。但需要注意的是，任何对ADCOF和ADCGN出厂设定值的修改都会降低ADC的线性性能。

对于系统失调误差校正，必须将ADC通道输入级连接到AGND上。用软件控制实现连续转换环，在转换过程中不断修改ADCOF的值，直到ADC转换结果数据(ADCDAT)读数从0变为1。如果ADCDAT值大于1，应该降低ADCOF中的值，直到ADCDAT读数从0变为1。这种数字化校正失调误差的分辨率可以达到0.25 LSB，范围可达到 V_{REF} 的 $\pm 3.125\%$ 。

对于系统增益误差校正，必须将ADC通道输入级连接到 V_{REF} 上。用软件控制实现连续转换环，在转换过程中不断修改ADCGN的值，直到ADCDAT读数从4094变为4095。如果ADCDAT值小于4094，应该降低ADCGN中的值，直到ADCDAT读数从4094变为4095。与失调误差校准一样，增益校准分辨率可以达到0.25 LSB，范围可达到 V_{REF} 的 $\pm 3\%$ 。

电流消耗

待机模式下，也就是上电但是没有转换情况下，ADC典型功耗为640 μA 。使用内部基准电压源时，电流要增加140 μA 。转换过程中，额外电流是0.3 μA 乘以采样频率(单位为kHz)。

时序

ADC时序如图16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。

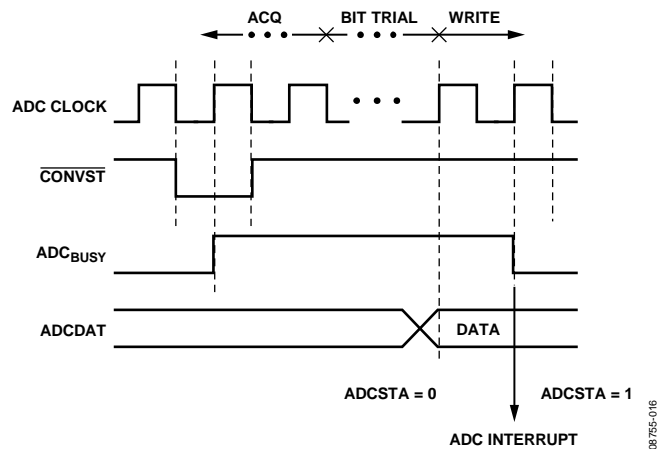


图16. ADC时序

ADuC7122

温度传感器

ADuC7122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ADuC7122芯片温度的近似指示器。

典型温度系数为 $-1.25 \text{ mV}/^{\circ}\text{C}$ 。

ADC寄存器接口

ADC通过许多寄存器进行控制和配置(参见表27)，参见表28至表34的说明。

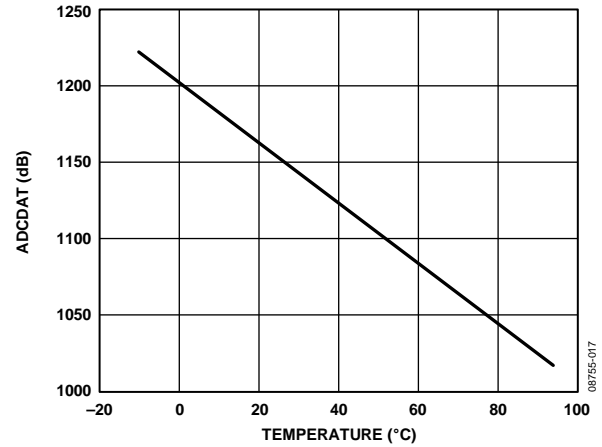


图17. ADC输出与温度的关系

表27. 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28)。
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。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进行转换时，也可以通过外部ADC _{Busy} 引脚读取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} 引脚变回低电平。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I/O”部分)。
ADCDAT	ADC数据结果寄存器。存放12位ADC转换结果数据。如表32所示。
ADCRST	ADC复位寄存器。用于将所有ADC寄存器复位至默认值。
PGA_GN	PADC0和PADC1的增益。

表28. ADCCON寄存器位功能描述(地址 = 0xFFFF0500, 默认值 = 0x0000A00)

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($f_{ADC} = f_{CORE}$, 转换 = 19 ADC时钟 + 采集时间)。 $f_{ADC}/1$ 。该分频器用于在外部时钟频率小于41.78 MHz时获得1 MSPS ADC。 $f_{ADC}/2$ (默认值)。 $f_{ADC}/4$ 。 $f_{ADC}/8$ 。 $f_{ADC}/16$ 。 $f_{ADC}/32$ 。
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	1 0	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。用户应将该位置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 μ s才能正确转换。 用户清0时, ADC进入掉电模式。
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。 CONVST引脚输入作为转换使能信号。 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。 保留 保留。

ADuC7122

表29. ADCCP寄存器位功能描述
(地址 = 0xFFFF0504, 默认值 = 0x00)

位	值	说明
7:5		保留
4:0		正向通道选择位
	00000	PADC0P
	00001	PADC1P
	00010	ADC0
	00011	ADC1
	00100	ADC2
	00101	ADC3
	00110	ADC4
	00111	ADC5
	01000	ADC6
	01001	ADC7
	01010	ADC8
	01011	ADC9
	01100	ADC10/AINCM
	01101	温度传感器
	01110	保留
	01111	保留
	10000	保留
	10001	保留
	10010	保留
	10011	IOVDD_MON
	10100	保留
	10101	保留
	10110	V _{REF}
	10111	AGND
	Others	保留

表30. ADCCN寄存器位功能描述
(地址 = 0xFFFF0508, 默认值 = 0x00)

位	值	说明
7:5		保留
4:0		反向通道选择位
	00000	PADC0N
	00001	PADC1N
	00010	ADC0
	00011	ADC1
	00100	ADC2
	00101	ADC3
	00110	ADC4
	00111	ADC5
	01000	ADC6
	01001	ADC7
	01010	ADC8
	01011	ADC9
	01100	ADC10/AINCM
	01101	保留
	01110	AGND
	01111	保留
	10000	I0GND
	其它	保留

表31. ADCSTA寄存器位功能描述
(地址 = 0xFFFF050C, 默认值 = 0x00)

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32. ADCDAT寄存器位功能描述
(地址 = 0xFFFF0510, 默认值 = 0x00000000)

位	值	说明
27:16		保存ADC结果(参见图15)。

表33. ADCRST寄存器位功能描述
(地址 = 0xFFFF0514, 默认值 = 0x00)

位	值	说明
0	1	用户置1时, 所有ADC寄存器复位为默认值。

表34. PGA_GN寄存器位功能描述
(地址 = 0xFFFF0520, 默认值 = 0x0000)

位	值	说明
15:12		保留。置0。
11:6		PADC0的PGA增益(PGA_PADC0_GN) = 1 + 4 X (PGA_ADC0_GN/32)
5:0		PADC1的PGA增益(PGA_PADC1_GN) = 1 + 4 X (PGA_ADC1_GN/32)

注意, PGA_PADC0_GN和PGA_PADC1_GN必须≤ 32。

转换器操作

这款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、伪差分模式和单端模式。

差模

ADuC7122包含一个基于两个容性DAC的逐次逼近型ADC。图18和图19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18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。

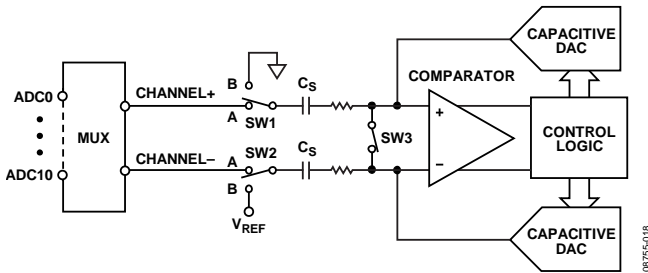


图18. ADC采集阶段

当ADC启动转换(见图19)时，SW3断开，而SW1和SW2移至位置B，使得比较器变得不平衡。转换开始时，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+} 和 V_{IN-} 输入的源输出阻抗一定要匹配，否则两个输入的建立时间不同会导致错误。差分模式的输入通道配置通过寄存器ADCCP和ADCCN设置。

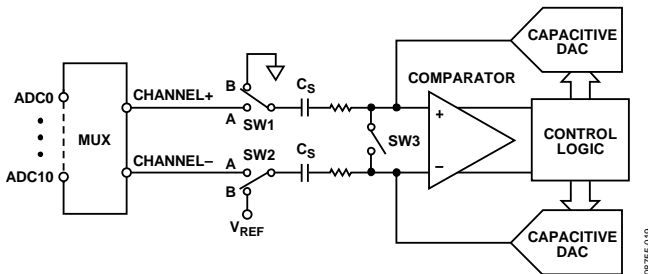


图19. ADC转换阶段

伪差分模式

在伪差分模式中，负通道(Channel-)连接到ADuC7122的 V_{IN-} 输入，SW2开关在A(V_{REF})和B(V_{IN-})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} 。在伪差分模式下，对于 V_{IN-} 通道，只应使能AINCM或PADCxN。ADCCN寄存器用于将Channel-设置为AINCM或PADCxN，Channel+可利用ADCCP寄存器来选择。

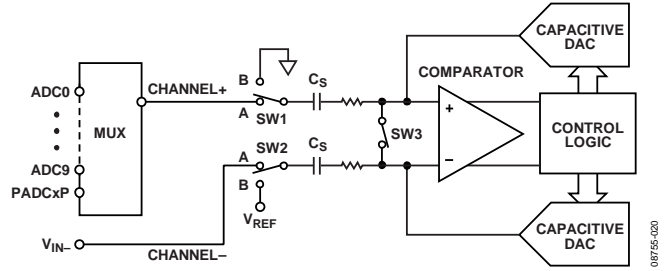


图20. 伪差分模式下ADC

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} 。ADuC7122有11个固定增益ADC通道和2个可编程增益ADC通道，这些通道可利用ADCCP寄存器使能。

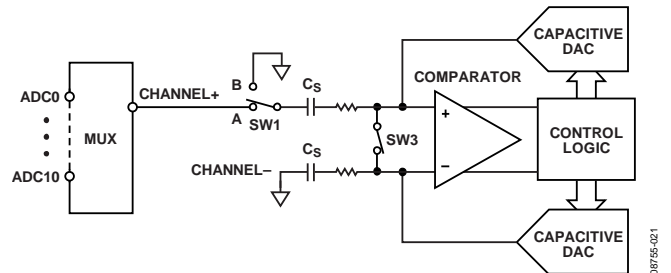


图21. 单端模式下ADC

模拟输入结构

ADC模拟输入结构等效电路如图22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22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 Ω 左右。电容C2为ADC采样电容，典型值为16 pF。

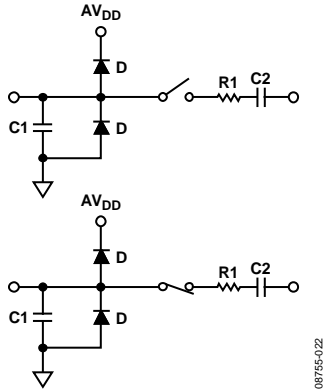


图22. 等效模拟输入电路转换阶段：开关打开，
采样阶段：开关闭合

在交流应用中，建议在相关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23和图24为ADC前端的示例。

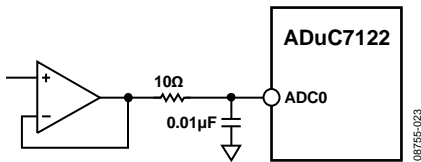


图23. 带缓冲的单端/伪差分输入

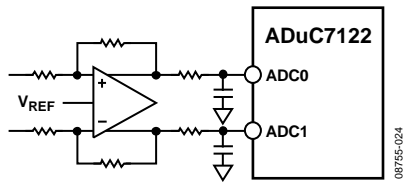


图24. 带缓冲的差分输入

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。在差分工作模式下，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35给出了不同 AV_{DD} 和 V_{REF} 条件下一些计算出的 V_{CM} 最小值和最大值。

表35. V_{CM} 范围

AV_{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

带隙基准电压源

ADuC7122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 $V_{REF-1.2}$ 和 $V_{REF-2.5}$ 引脚上。当采用内部基准电压源时， $V_{REF-1.2}$ 与AGND之间必须接上一个470 nF的电容， $V_{REF-2.5}$ 与AGND之间也必须接上一个470 nF的电容，以确保ADC转换时的稳定性和快速响应。

带隙基准电压源也通过缓冲连接到BUF_VREF1和BUF_VREF2引脚，用作系统中其它电路的基准电压源。这些引脚需连接至少0.1 μF的电容，以降低噪声。

带隙基准电压接口包括一个8位REFCON寄存器，如表36所示。采用内部基准电压源执行ADC或DAC转换时，建议使用能REFCON的位0和位1。

ADC转换可采用外部基准电压源。要采用外部2.5 V基准电压源执行ADC转换，应清除REFCON[1]，并将该外部基准源施加于VREF_2.5引脚。要采用外部1.2 V基准电压源，应清除REFCON[0]，并将该外部基准源施加于 $V_{REF-1.2}$ 引脚。注意，将该外部基准源施加于 $V_{REF-1.2}$ 引脚时，会在内部影响2.5 V基准源，因为2.5 V基准源是从1.2 V基准源产生。

电源监控器

当IOV_{DD}电源引脚电压降到两个电源跳变点之一以下时，ADuC7122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，该位会立即清0。注意，如果在CMP变为高电平(IOV_{DD}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。用户应确保在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。

当ADC通道选择位配置为IOVDD_MON (ADCCP[4:0] = 10011)时，ADC可以转换电源监控比较器输入端的电压。测量内部通道时，必须使能内部缓冲器。对任何内部通道进行采样时，应使能内部缓冲器以隔离外部干扰。测量此电压之前，需要执行以下步骤：

1. 利用ADC测量VREF。
2. 设置ADCCP = IOVDD_MON通道。
3. 设置60 μs的典型延迟时间。
4. 在IOVDD_MON通道上执行ADC转换(ADCCON使用值0x2AA3以获得最佳结果)。

ADC多路复用器选择切换与转换启动之间的延迟时间用于让ADC采样电容上的电压建立到分压后的电源电压。

表36. REFCON寄存器位功能描述(地址 = 0xFFFF0480, 默认值 = 0x01)

位	说明
7:1	保留。
2	保留。始终置为1。此位用于将内部2.5 V基准源的缓冲版本输出到BUF_VREF1和BUF_VREF2上。要禁用此缓冲器，用户必须清除REFCON (= 0x00)以禁用内部基准源。
1	内部2.5V基准电压输出使能。 用户置1时，内部2.5V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。要将外部基准电压源连接到V _{REF-2.5} 引脚，此位也应清0。
0	内部1.2V基准电压输出使能。 用户置1时，内部1.2V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。

表37. PSMCON寄存器位功能描述(地址 = 0xFFFF0440; 默认值 = 0x08或0x00, 取决于器件电源电平)

位	名称	说明
7:4	保留	保留位。清0。
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IOV _{DD} 电压高于所选跳变点或PSM处于掉电模式。 该位为0表示IOV _{DD} 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ADuC7122置1，表明I/O电源电压偏低。PSMI位可用来中断处理器。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

ADuC7122片内集成Flash/EE存储器技术，向用户提供非易失、在线可编程存储空间。

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ADuC7122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

ADuC7122内置两个64 kB的Flash/EE存储器阵列。在闪存模块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。在闪存模块1中，所有64 kB的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

ADuC7122的Flash/EE存储器阵列完全满足两大Flash/EE存储器关键特性要求：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个独立的、连续的事件：

1. 初始页面擦除时序
2. 读/校验单个Flash/EE时序
3. 存储器的字节编程时序
4. 二次读/校验耐久性周期时序

在进行可靠性验证时，要测试各Flash/EE存储器模块的三个不同页块。各Flash/EE存储器模块顶部、中部和底部的完整Flash/EE页面循环存取10,000次(从0x0000至0xFFFF)。

如“概述”部分所示，Flash/EE存储器耐久性是根据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。这个结果满足工业温度范围内不同电源电压下最小耐久性达到10,000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。器件在特定结温($T_j = 85^\circ\text{C}$)下根据标准JEDEC保持期限规格(A117)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25所示。

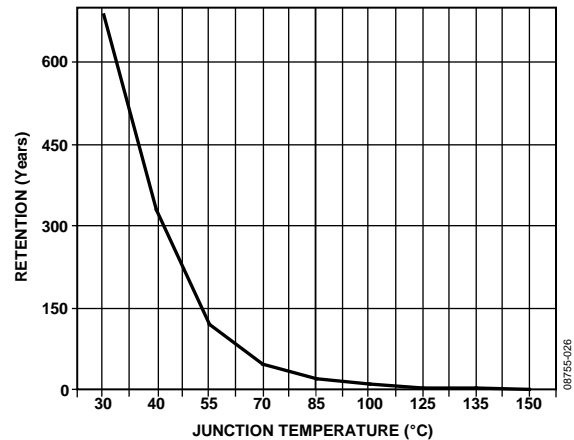


图25. Flash/EE存储器数据保持期限

串行下载(在线编程)

ADuC7122可利用I²C串行端口下载代码。如果BM引脚通过一个1 kΩ的外部下拉电阻下拉到低电平，ADuC7122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引脚被拉低，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，用户可以向Flash/EE的整个126 kB Flash/EE存储器空间内下载代码(尽管该器件在其目标应用硬件中为在线状态)。开发系统提供PC可执行串行下载程序和硬件适配板，支持通过I²C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或串行下载器读取。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FEEExHID寄存器来设定或取消保护。复位后，保护失效。

可通过写入FEEExPRO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。FEEExPRO寄存器采用一个密钥保护来避免直接访问。一旦密钥被保存，必须再次输入此密钥才可更改FEEExPRO。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FEEExPRO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FEEExPRO寄存器。

密钥写入时序

1. 写入FEEExPRO寄存器的某一位，相应页面会被保护；
2. 将FEEExMOD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FEEExADR和FEEExDAT寄存器；
4. 在FEEExCON寄存器中运行写密钥命令0x0C，通过监测FEEExSTA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FEEExPRO值。如果密钥选择的值是0xDEADDEAD，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=0xFFFFFFFF; //Protect pages 4 to 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FEEExADR = 0xDEAD且FEEExDAT = 0xDEADDEAD。

FLASH/EE控制接口

表38. FEE0DAT寄存器

名称	地址	默认值	访问类型
FEE0DAT	0xFFFF0E0C	0xFFFF	R/W

FEE0DAT是一个16位数据寄存器。

表39. FEE0ADR寄存器

名称	地址	默认值	访问类型
FEE0ADR	0xFFFF0E10	0x0000	R/W

FEE0ADR是一个16位地址寄存器。

表40. FEE0SGN寄存器

名称	地址	默认值	访问类型
FEE0SGN	0xFFFF0E18	0xFFFFFFFF	R

FEE0SGN为24位编码签名。

表41. FEE0PRO寄存器

名称	地址	默认值	访问类型
FEE0PRO	0xFFFF0E1C	0x00000000	R/W

FEE0PRO在随后的复位后提供保护。它需要一个软件密钥(见表57)。如上所述，FEEExPRO寄存器从位30到0的各位保护2 kB的存储器模块，即设置位0为低电平可保护页0至页3，设置位2为低电平可保护页8至页11。

表42. FEE0HID寄存器

名称	地址	默认值	访问类型
FEE0HID	0xFFFF0E20	0xFFFFFFFF	R/W

FEE0HID提供立即保护。它不需要任何软件密钥(见表57)。

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
```

ADuC7122

表43. FEE1DAT寄存器

名称	地址	默认值	访问类型
FEE1DAT	0xFFFF0E8C	0XXXXX	R/W

FEE1DAT是一个16位数据寄存器。

表44. FEE1ADR寄存器

名称	地址	默认值	访问类型
FEE1ADR	0xFFFF0E90	0x0000	R/W

FEE1ADR是一个16位地址寄存器。

表45. FEE1SGN寄存器

名称	地址	默认值	访问类型
FEE1SGN	0xFFFF0E98	0FFFFFFF	R

FEE1SGN为24位编码签名。

表46. FEE1PRO寄存器

名称	地址	默认值	访问类型
FEE1PRO	0xFFFF0E9C	0x00000000	R/W

FEE1PRO在随后的复位后提供保护。它需要一个软件密钥(见表58)。

表47. FEE1HID寄存器

名称	地址	默认值	访问类型
FEE1HID	0xFFFF0EA0	0xFFFFFFFF	R/W

FEE1HID提供立即寄存器保护。它不需要任何软件密钥(见表58)。

表48. FEE0STA寄存器

名称	地址	默认值	访问类型
FEE0STA	0xFFFF0E00	0x0000	R

表49. FEE1STA寄存器

名称	地址	默认值	访问类型
FEE1STA	0xFFFF0E80	0x0000	R

表50. FEE0MOD寄存器

名称	地址	默认值	访问类型
FEE0MOD	0xFFFF0E04	0x80	R/W

表51. FEE1MOD寄存器

名称	地址	默认值	访问类型
FEE1MOD	0xFFFF0E84	0x80	R/W

表52. FEE0CON寄存器

名称	地址	默认值	访问类型
FEE0CON	0xFFFF0E08	0x0000	R/W

表53. FEE1CON寄存器

名称	地址	默认值	访问类型
FEE1CON	0xFFFF0E88	0x0000	R/W

表54. FEEExSTA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FEEExMOD寄存器的Flash/EE中断使能位置1时，自动置1。 读取FEEExSTA寄存器后，该位清0。
2	Flash/EE控制器繁忙。 当控制器忙时，该位自动置1； 当控制器空闲时，该位自动清0。
1	命令失败。 当命令的执行失败时，该位自动置1； 读取FEEExSTA寄存器后，该位自动清0。
0	命令完成。 当命令完成时，ADuC7122将该位置1。 读取FEEExSTA寄存器后，该位自动清0。

表55. FEEExMOD寄存器位功能描述

位	说明
7:5	保留。这些位应始终设置为0(写闪存控制密钥的情况除外)。
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。用户应始终将该位置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。

表56. FEEExCON中的命令代码

代码	命令	说明
0x00 ¹	零点	空闲状态。
0x01 ¹	单次读取	将FEEExADR索引的16位数据载入FEEExDAT。
0x02 ¹	单次写入	在FEEExADR指向的地址处写入FEEExDAT。此操作耗时50 μs。
0x03 ¹	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 ¹	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 ¹	单次擦除	擦除FEEExADR索引的页。
0x06 ¹	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。

¹ 执行完这些命令后，FEEExCON寄存器总是立即读取0x07。

表57. 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。 用户将该位置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。

表58. 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。 用户将该位置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。

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中获得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。这与SRAM取指令相反，当CD位=0时，从SRAM执行指令能够在一个周期内完成。根据指令不同，无论CD位为何值，在访问数据之前可能需要一定的死区时间。

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59所示。

表59. 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

$1 < N \leq 16$ ，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异常矢量全部位于存储器阵列的底部，从地址0x00000000到地址0x00000020，如图26所示。

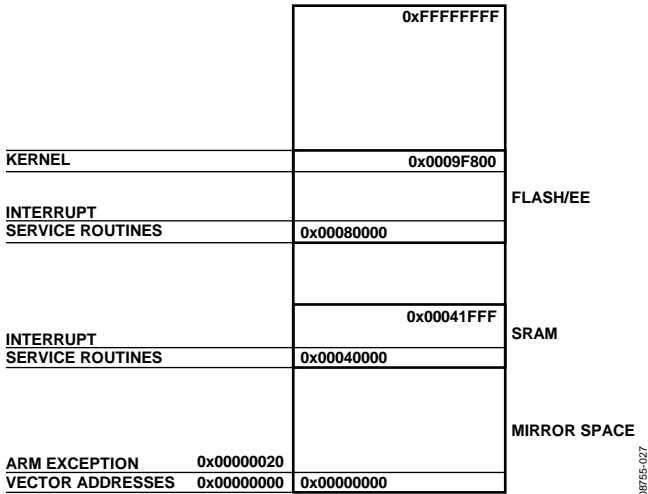


图26. 异常重映射

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

ADuC7122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ADuC7122在正常模式工作(BM引脚为高电平)，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，便可配置从Flash/EE执行存储器重映射。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。

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，当前为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60. 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61. 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。

ADuC7122

RSTCFG寄存器

名称: RSTCFG
地址: 0xFFFF024C
默认值: 0x00
访问类型: 读/写

表62. RSTCFG寄存器位功能描述

位	说明
7至3	保留。始终置为0。
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。

RSTKEY1寄存器

名称: RSTKEY1
地址: 0xFFFF0248
默认值: 不适用
访问类型: 写操作

RSTKEY2寄存器

名称: RSTKEY2
地址: 0xFFFF0250
默认值: 不适用
访问类型: 写操作

表63. RSTCFG写序列

名称	代码
RSTKEY1	0x76
RSTCFG	用户设定值
RSTKEY2	0xB1

其他模拟外设

DAC

ADuC7122片内集成12个12位缓冲电压输出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 Ω /100 pF。

每个DAC都有两个可选电压输出范围：0 V至 V_{REF} （内部2.5 V带隙基准电压源）和0 V至 AV_{DD} 。最大信号范围为0 V至 AV_{DD} 。

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。本部分介绍DACxCON和DACxDAT（参见表64至表67）的具体功能。

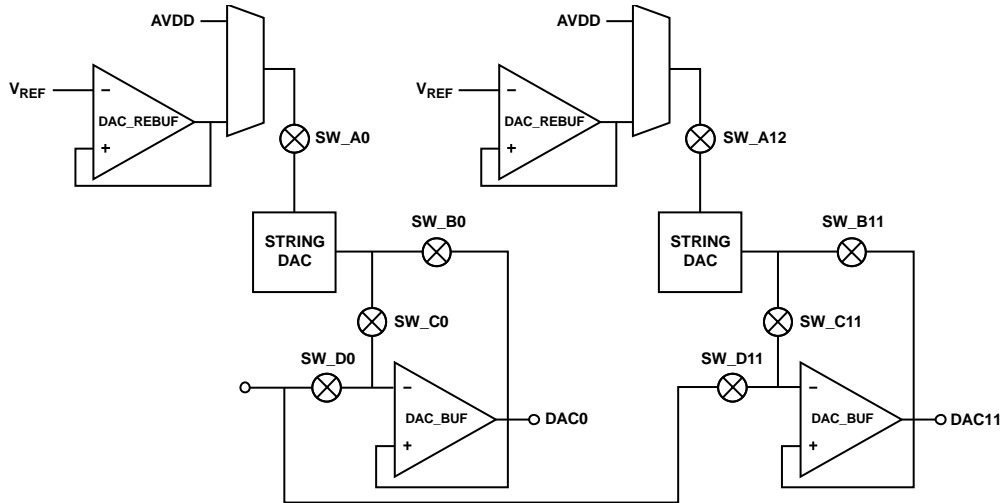


图27. DAC配置

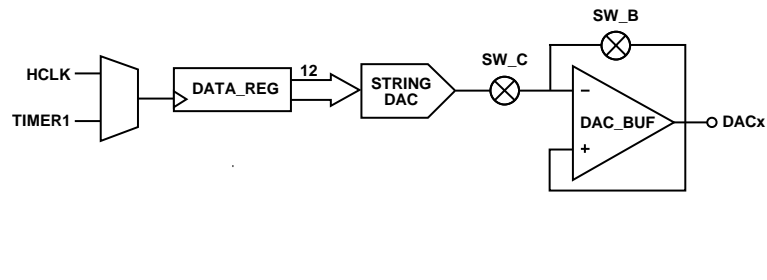


图28. DAC用户功能

表64. DACxCON寄存器

名称	地址	默认值	访问类型
DAC0CON	0xFFFF0580	0x100	R/W
DAC1CON	0xFFFF0588	0x100	R/W
DAC2CON	0xFFFF0590	0x100	R/W
DAC3CON	0xFFFF0598	0x100	R/W
DAC4CON	0xFFFF05A0	0x100	R/W
DAC5CON	0xFFFF05A8	0x100	R/W
DAC6CON	0xFFFF05B0	0x100	R/W
DAC7CON	0xFFFF05B8	0x100	R/W
DAC8CON	0xFFFF05C0	0x100	R/W
DAC9CON	0xFFFF05C8	0x100	R/W
DAC10CON	0xFFFF05D0	0x100	R/W
DAC11CON	0xFFFF05D8	0x100	R/W

ADuC7122

表65. DACxCON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		保留。
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		保留。
2	0		保留。始终清0。
1:0		DACRNx	保留。始终清0。
	00		V _{REF} /AGND。
	01		保留。
	10		保留。
	11		AV _{DD} /AGND。

表66. DACxDAT寄存器

名称	地址	默认值	访问类型
DAC0DAT	0xFFFF0584	0x00000000	R/W
DAC1DAT	0xFFFF058C	0x00000000	R/W
DAC2DAT	0xFFFF0594	0x00000000	R/W
DAC3DAT	0xFFFF059C	0x00000000	R/W
DAC4DAT	0xFFFF05A4	0x00000000	R/W
DAC5DAT	0xFFFF05AC	0x00000000	R/W
DAC6DAT	0xFFFF05B4	0x00000000	R/W
DAC7DAT	0xFFFF05BC	0x00000000	R/W
DAC8DAT	0xFFFF05C4	0x00000000	R/W
DAC9DAT	0xFFFF05CC	0x00000000	R/W
DAC10DAT	0xFFFF05D4	0x00000000	R/W
DAC11DAT	0xFFFF05DC	0x00000000	R/W

表67. 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	保留。
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29所示。

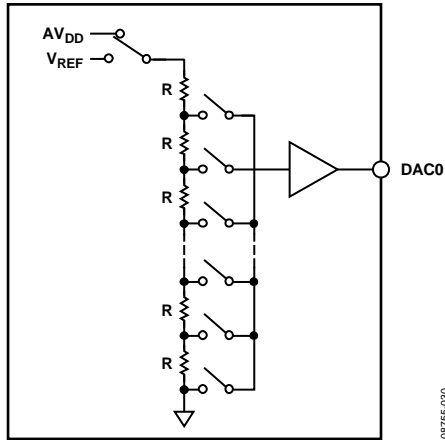


图29. DAC结构

如图29所示，用户可在软件中选择各DAC的基准电压源。基准电压源可为AV_{DD}或V_{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 kΩ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30反映了这种效应的一般表现(失调和增益误差忽略不计)。图30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30仅代表输入范围为0至AV_{DD}模式时的传递函数。在0 V至V_{REF}模式(V_{REF} < AV_{DD})下，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。

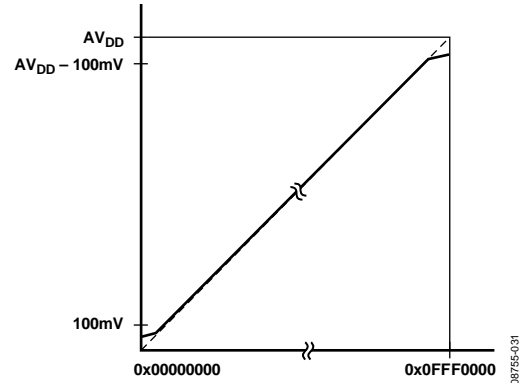


图30. 放大器饱和引起的端点非线性

当输出有负载时，图30中的端点非线性会变得更差。ADuC7122数据手册中的技术参数是在DAC输出端接有一个接地的5 kΩ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30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。

通过写入RSTCFG寄存器，可以配置DAC在看门狗或软件复位后保留输出电压。

LDO(低压差稳压器)

ADuC7122集成一个LDO，它从IOVDD电源产生约2.6 V的内核电源电压(LVDD)。由于LDO是从IOVDD驱动，因此IOVDD电源电压需要大于2.7 V。

必须将一个低等效串联电阻(ESR)的0.47 μF外部补偿电容(C_T)紧挨着LVDD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LVDD电压仅用于为ADuC7122提供电源。因此，用户不应将LVDD引脚用作其他芯片的电源。此外，IOVDD引脚应具有良好的电源去耦，以便提高LDO的线性稳压性能。

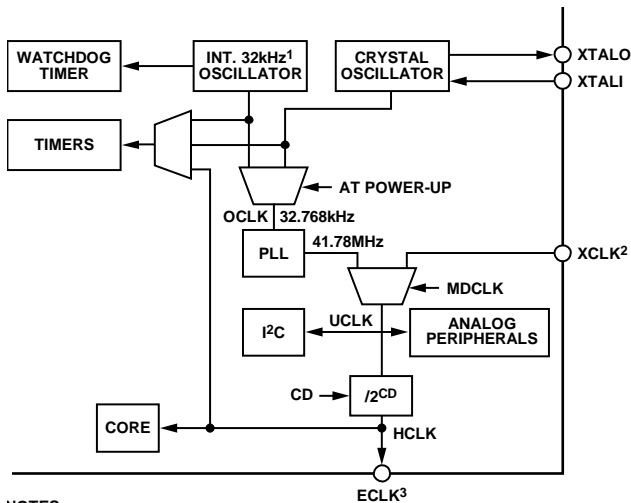
LVDD引脚无电池反接、限流或热关断保护；因此，在正常工作或电路板制造期间，ADuC7122的用户任何时候都不得将此引脚短接至地。

振荡器和锁相环—电源控制

ADuC7122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ECLK引脚输出，如图31所示。注意，当使用ECLK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。

ADuC7122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75)和POWCON(参见表76)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。



NOTES
 1. 32.768kHz \pm 3%.
 2. TO USE THE SECONDARY FUNCTION OF P1.4 AS XCLK, PLLCON BITS[1:0] MUST EQUAL 11.
 3. WHEN THE SECONDARY FUNCTION FOR P1.4 IS SET TO 2 (THAT IS, GP1CON[17:16] = 10), THE ECLK FUNCTION IS SELECTED BY DEFAULT.

图31. 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 μ s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。

源代码示例

```
T2LD = 5;
T2CON = 0x480;

while ((T2VAL == t2val_old) || (T2VAL >
3)) //ensures timer value loaded

  IRQEN = 0x10;
//enable T2 interrupt

  PLLKEY1 = 0xAA;
  PLLCON = 0x01;
  PLLKEY2 = 0x55;

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

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进行测试，判断有没有来自看门狗定时器的复位信号。

外部时钟选择

要切换到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，容差为1%。

源代码示例

```
T2LD = 5;
T2CON = 0x480;

while ((T2VAL == t2val_old) || (T2VAL >
3)) //ensures timer value 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

ADuC7122支持多种工作模式选项。表68说明了不同模式下ADuC7122的哪些模块上电，并给出了上电时间。表69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68. 工作模式

模式	内核	外设	PLL	XTAL/定时器2/定时器3	XIRQ	启动/上电时间
有效	开	开	开	开	开	130 ms(CD = 0)
暂停		开	开	开	开	24 ns (CD = 0); 3.06 μ s (CD = 7)
浅休眠			开	开	开	24 ns (CD = 0); 3.06 μ s (CD = 7)
休眠				开	开	1.58 ms
停止					开	1.7 ms

表69. 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	30	21.2	13.8	11	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	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
100	停止	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25

ADuC7122

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74)。

表70. PLLKEYx寄存器

名称	地址	默认值	访问类型
PLLKEY1	0xFFFF0410	0x0000	W
PLLKEY2	0xFFFF0418	0x0000	W

表71. PLLCON寄存器

名称	地址	默认值	访问类型
PLLCON	0xFFFF0414	0x21	R/W

表72. POWKEYx寄存器

名称	地址	默认值	访问类型
POWKEY1	0xFFFF0404	0x0000	W
POWKEY2	0xFFFF040C	0x0000	W

表73. POWCON寄存器

名称	地址	默认值	访问类型
POWCON	0xFFFF0408	0x03	R/W

表74. PLLCON和POWCON写时序

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

表75. PLLCON寄存器位功能描述

位	值	名称	说明
7:6			保留。
5	1	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。
	0		用户清0时，使用外部32 kHz晶振。
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引脚上的外部时钟。

表76. 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。
	000		有效模式。
	001		暂停模式。
	010		浅休眠。
	011		休眠模式。IRQ0至IRQ3和定时器2 可以唤醒ADuC7122。
	100		停止模式。
	其它		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。

数字外设

PWM概述

ADuC7122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H桥控制电机是关闭的。在标准PWM模式下，从三对PWM引脚内输出信号。用户既可以控制每一对输出引脚的工作周期，又可以单独控制每一个输出端的占空比。

表77. 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

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32所示。

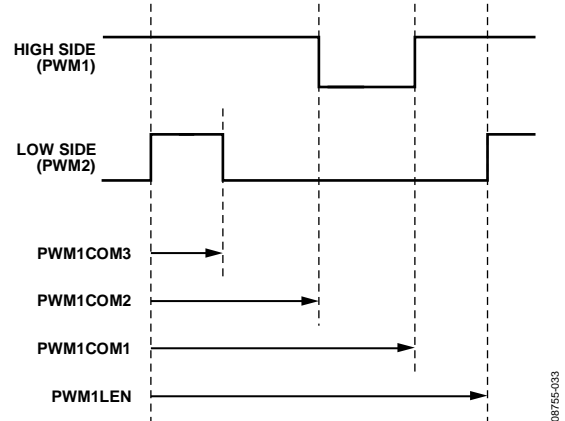


图32. PWM时序

PWM时钟可通过PWMCON1选择，UCLK分频值如下：2、4、8、16、32、64、128或256。PWMxLEN用于设定PWM周期长度。

PWM波形由16位定时器的计数值和比较寄存器的内容来决定，如图32所示的PWM1和PWM2波形。

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。

在H桥模式下，HMODE = 1，并通过表78确定PWM输出。

表78. 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SYNC引脚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SYNC引脚上发生的跃迁。
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	PWM4NV	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。 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。 如果HOFF = 1且HMODE = 1，请参见表79。 如果不是在H桥模式下，此位无作用。

ADuC7122

位	名称	说明
8:6	PWMCP[2:0]	PWM时钟预分频器位。设置UCLK分频数。 000 = UCLK/2。 001 = UCLK/4。 010 = UCLK/8。 011 = UCLK/16。 100 = UCLK/32。 101 = UCLK/64。 110 = UCLK/128。 111 = UCLK/256。
5	POINV	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1至位5。 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。

表79. PWM输出选择¹

PWMCON1寄存器				PWM输出			
ENA	HOFF	POINV	DIR	PWM1	PWM2	PWM3	PWM4
0	0	X	X	1	1	1	1
X	1	X	X	1	0	1	0
1	0	0	0	0	0	HS	LS
1	0	0	1	HS	LS	0	0
1	0	1	0	HS	LS	1	1
1	0	1	1	1	1	HS	LS

¹ X = 无关; HS = 高端; LS = 低端。

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80)。

表80. 比较寄存器

名称	地址	默认值	访问类型
PWM1COM1	0xFFFF0F84	0x00	R/W
PWM1COM2	0xFFFF0F88	0x00	R/W
PWM1COM3	0xFFFF0F8C	0x00	R/W
PWM2COM1	0xFFFF0F94	0x00	R/W
PWM2COM2	0xFFFF0F98	0x00	R/W
PWM2COM3	0xFFFF0F9C	0x00	R/W
PWM3COM1	0xFFFF0FA4	0x00	R/W
PWM3COM2	0xFFFF0FA8	0x00	R/W
PWM3COM3	0xFFFF0FAC	0x00	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81. PWMCON2寄存器位功能描述
(地址 = 0xFFFF0FB4, 默认值 = 0x00)

位	值	名称	说明
7	1	CSEN	转换开始使能。 用户置1时, 使能PWM产生转换开始信号。
	0		用户清0时, 禁用PWM转换开始信号。
3:0	0000	CSD3	用户清0时, 禁用PWM转换开始信号。 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。图33中的例子是延迟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33)。

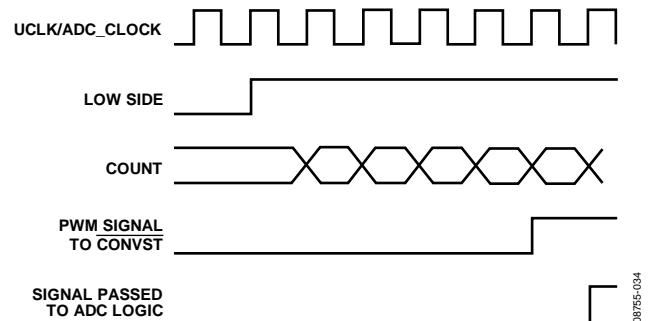


图33. ADC转换

通用I/O

ADuC7122提供32个双向通用输入/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84)。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。

表82. GPxCON寄存器

名称	地址	默认值	访问类型
GP0CON	0xFFFF0D00	0x00000000	R/W
GP1CON	0xFFFF0D04	0x00000000	R/W
GP2CON	0xFFFF0D08	0x00000000	R/W
GP3CON	0xFFFF0D0C	0x11111111	R/W

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。

ADuC7122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84所示。

表83. GPxCON寄存器位功能描述

位	说明
31:30	保留
29:28	Px.7引脚的功能选择
27:26	保留
25:24	Px.6引脚的功能选择
23:22	保留
21:20	Px.5引脚的功能选择
19:18	保留
17:16	Px.4引脚的功能选择
15:14	保留
13:12	Px.3引脚的功能选择
11:10	保留
9:8	Px.2引脚的功能选择
7:6	保留
5:4	Px.1引脚的功能选择
3:2	保留
1:0	Px.0引脚的功能选择

表84. GPIO引脚功能描述¹

端口	引脚	配置(参见GPxCON)			
		00	01	10	11
0	P0.0	GPIO	SCL1	不适用	PLAI[5]
	P0.1	GPIO	SDA1	不适用	PLAI[4]
	P0.2	GPIO	SPICLK	ADC _{BUSY}	PLAO[13]
	P0.3	GPIO	SPIMISO	SYNC	PLAO[12]
	P0.4	GPIO	SPIMOSI	TRIP	PLAI[11]
	P0.5	GPIO	SPIC _S	CONVST	PLAI[10]
	P0.6	GPIO	MRST	不适用	PLAI[2]
	P0.7	GPIO	TRST	不适用	PLAI[3]
1 ²	P1.0	GPIO	SIN	SCL2	PLAI[7]
	P1.1	GPIO	SOUT	SDA2	PLAI[6]
	P1.4	GPIO	PWM1	ECLK/XCLK	PLAI[8]
	P1.5	GPIO	PWM2	不适用	PLAI[9]
	P1.6	GPIO	不适用	不适用	PLAO[5]
	P1.7	GPIO	不适用	不适用	PLAO[4]
	2	P2.0	GPIO/IRQ0	不适用	不适用
P2.1		GPIO/IRQ1	不适用	不适用	PLAI[12]
P2.2		GPIO	不适用	不适用	PLAI[1]
P2.3		GPIO/IRQ2	不适用	不适用	PLAI[14]
P2.4		GPIO	PWM5	不适用	PLAO[7]
P2.5		GPIO	PWM6	不适用	PLAO[6]
P2.6		GPIO/IRQ3	不适用	不适用	PLAI[15]
P2.7		GPIO	不适用	不适用	PLAI[0]
3	P3.0	GPIO	N/A	不适用	PLAO[0]
	P3.1	GPIO	不适用	不适用	PLAO[1]
	P3.2	GPIO/IRQ4	PWM3	不适用	PLAO[2]
	P3.3	GPIO/IRQ5	PWM4	不适用	PLAO[3]
	P3.4	GPIO	不适用	不适用	PLAO[8]
	P3.5	GPIO	不适用	不适用	PLAO[9]
	P3.6	GPIO	不适用	不适用	PLAO[10]
P3.7/BM	GPIO	不适用	不适用	PLAO[11]	

¹ N/A(不适用)表示不存在第二功能。

² 切勿写入P1.2或P1.3。

表85. GPxPAR寄存器

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

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。

表86. 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
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

表87. GPxDAT寄存器

名称	地址	默认值	访问类型
GP0DAT	0xFFFF0D20	0x000000XX	R/W
GP1DAT	0xFFFF0D30	0x000000XX	R/W
GP2DAT	0xFFFF0D40	0x000000XX	R/W
GP3DAT	0xFFFF0D50	0x000000XX	R/W

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。

表88. 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)。

表89. GPxSET寄存器

名称	地址	默认值	访问类型
GP0SET	0xFFFF0D24	0x000000XX	W
GP1SET	0xFFFF0D34	0x000000XX	W
GP2SET	0xFFFF0D44	0x000000XX	W
GP3SET	0xFFFF0D54	0x000000XX	W

表90. 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。 用户清0不会影响数据输出。
15:0	保留。

GPxSET是端口x的数据设置寄存器。

表91. GPxCLR寄存器

名称	地址	默认值	访问类型
GP0CLR	0xFFFF0D28	0x000000XX	W
GP1CLR	0xFFFF0D38	0x000000XX	W
GP2CLR	0xFFFF0D48	0x000000XX	W
GP3CLR	0xFFFF0D58	0x000000XX	W

GPxCLR是端口x的数据清除寄存器。

表92. 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。

下列GPIO引脚提供开集功能：P1.7、P1.6、P2.x和P3.x。开集功能可利用GP1OCE[7:6]、GP2OCE[7:0]和GP3OCE[7:0]配置。

表93. 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

ADuC7122有一个16450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且支持网络可寻址模式。

ADuC7122的P1.0和P1.1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

ADuC7122提供两种UART波特率生成模式：正常450 UART波特率生成模式和ADuC7122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：

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

表94列举了几种常见波特率值。

表94.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%

ADuC7122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。

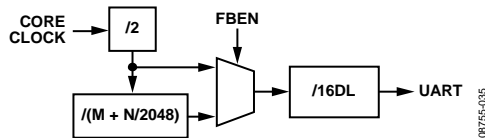


图34. 波特率生成选项

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，

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{19,200 \times 16 \times 67 \times 2}$$

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

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、COMRX和COMIEN0。当寄存器COMCON0的位7为1时，可以访问COMDIVx。

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

ADuC7122

表95. 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TxD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 to 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96. COMCON1寄存器位功能描述

位	名称	说明
7:5		保留位。未使用。
4	Loopback	用户置1, 使能回送模式。 此时强制TxD为高电平。
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。

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97. 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SIN保持低电平超过最大字长时置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。

ADuC7122

UART中断使能寄存器0

名称: COMIEN0
 地址: 0xFFFF0804
 默认值: 0x00
 访问类型: 读/写
 功能: 该8位寄存器用于使能或禁用单独的UART中断源。

表98. COMIEN0寄存器位功能描述

位	名称	说明
7:4		保留。未使用。
3	EDSSI	调制解调器状态中断使能位。用户置1，在COMSTA0[3:1]的任一位被置1时使能中断产生。由用户清0。
2	ELSI	RxD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

名称: COMIID0
 地址: 0xFFFF0808
 默认值: 0x01
 访问类型: 只读
 功能: 该8位寄存器用于反映UART中断源。

表99. 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ADuC7122小数分频器操作。

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

表100. 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94。
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94。

I²C

ADuC7122集成两个I²C外设，用户可将这些设备单独配置成完全I²C兼容型I²C总线主机或者完全I²C兼容型总线从机。这两个外设完全相同，因此这里仅介绍一个。

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。

I²C总线系统的外设地址由用户编程设定。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，使其响应四个从机地址。

I²C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；在初始地址传输期间，主机发送从机的地址和数据传输方向(读和/或写)。如果主机没有输掉仲裁且从机应答了最后一个字节，那么开始向从机传输数据。传输会持续到主机发送一个停止条件为止，然后总线进入空闲状态。

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

ADuC7122的I²C接口具备以下特性：

- 支持重复起始条件。在主机模式下，通过对ADuC7122进行编程，可以允许其重复启动一次。在从机模式下，ADuC7122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I²C主机模式下，ADuC7122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ADuC7122带来任何问题。不过，ADuC7122无法使能时钟延展。
- 在从机模式下，通过对ADuC7122进行编程，可以允许其返回一个非应答信号(NACK)。这样可以保证在I²C数据传输结束时校验和字节是有效的。
- 在主机模式下，支持总线仲裁。
- 支持内、外部回送，用于I²C硬件测试。在回送模式下，
- 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。

I²C功能的外部引脚配置

ADuC7122器件的I²C引脚为用于I2C0的P0.0和P0.1，以及用于I2C1的P1.0和P1.1。

P0.0和P1.0的功能是传输I²C时钟信号，而P0.1和P1.1的功能是传输I²C数据信号。例如，要配置I2C0引脚(SCL1和SDA1)，GP0CON寄存器的位0和位4必须置为1，以使能I²C模式。或者，要配置I2C1引脚(SCL2和SDA2)，GP1CON寄存器的位1和位5必须置为1，以使能I²C模式。

串行时钟生成

系统中的I²C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。

I2CxDIV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是时钟高电平周期。

DIVL是时钟低电平周期。

因而，如果希望串行时钟为100 kHz，那么

$$\text{应该配置 } DIVH = DIVL = 0xCF$$

如果希望串行时钟为400 kHz，那么

$$\text{应该配置 } DIVH = 0x28, \quad DIVL = 0x3C$$

I2CxDIV寄存器对应于DIVH:DIVL。

I²C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)被忽略。

ADuC7122还支持10位寻址模式。当I2CxSCTL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I²C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。

ADuC7122

主机模式

在主机模式下，I2CADR0寄存器用于存储器件的I²C地址。

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

在10位地址模式下，10位地址创建如下：

I2CADR0[7:3]的值必须为11110b。

I2CADR0[2:1] = 地址位[9:8]。

I2CADR1[7:0] = 地址位[7:0]。

I2CADR0[0]是读/写位。

I²C寄存器

I²C外设接口由数个寄存器组成，这些寄存器将在以下部分中讨论。

I²C主机寄存器

I²C主机控制寄存器

名称： I2C0MCTL, I2C1MCTL

地址： 0xFFFF0880, 0xFFFF0900

默认值： 0x0000, 0x0000

访问类型： 读/写

功能： 该16位寄存器用于在主机模式下配置I²C外设。

表101. I2CxMCTL寄存器位功能描述

位	名称	名称
15:9		保留。这些位为保留位，不允许向其中写入数据。
8	I2CMCENI	I ² C发送完成中断使能位。 将该位置1，则当从I ² C总线上检测到停止条件时，使能中断。 将该位清0，则禁用中断源。
7	I2CNACKENI	I ² C NACK接收中断使能位。 将该位置1，则当I ² C主机接收到NACK时，使能中断。 将该位清0，则禁用中断源。
6	I2CALENI	I ² C仲裁失效中断使能位。 将该位置1，则当I ² C主机没有获得对I2C总线的控制权时，使能中断。 将该位清0，则禁用中断源。
5	I2CMTENI	I ² C发送中断使能位。 将该位置1，则当I ² C主机完成一个字节的传输后，使能中断。 将该位清0，则禁用中断源。
4	I2CMRENI	I ² C接收中断使能位。 将该位置1，则当I ² C主机接收到数据时，使能中断。 用户清0，则在I ² C主机接收数据的过程中，禁用中断。
3		保留。应在此位写入0值。
2	I2CILEN	I ² C内部回送使能位。 该位置1，可以使能回送测试模式。在该模式下，SCL和SDA信号在内部分别与各自的输入信号相连。 用户清0，可禁用回送模式。
1	I2CBD	I ² C主机撤回禁用位。 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，则等到I ² C总线释放后执行撤回。
0	I2CMEN	I ² C主机使能位。 用户置1，使能I ² C主机模式。 用户清0，禁用I ² C主机模式。

I²C主机状态寄存器

名称: I2C0MSTA, I2C1MSTA
 地址: 0xFFFFF0884, 0xFFFFF0904
 默认值: 0x0000, 0x0000
 访问类型: 读操作
 功能: 该16位寄存器是主机模式下的I²C状态寄存器。

表102.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 ² C总线忙碌状态位。 当从I ² C总线上检测到起始条件后, 该位被置1。 当从I ² C总线上检测到停止条件后, 该位被清0。
9	I2CMRxFO	主机接收FIFO溢出。 接收FIFO已满后, 又有一个字节的数据写入FIFO时, 该位被置1。 在其它条件下, 该位被清0。
8	I2CMTC	I ² C发送完成状态位。 当主机与从机之间完成一次传输后, 该位被置1。如果I2CxMCTL寄存器的I2CMCENI位被置1, 当该位的值为1时, 使能中断。 将该位清0, 则禁用中断源。
7	I2CMNA	I ² C主机NACK数据位。 在执行数据写传输过程中, 当主机接收到一个NACK条件后, 该位被置1。如果I2CxMCTL寄存器的I2CNACKENI位被置1, 当该位的值为1时, 使能中断。 在其它条件下, 该位被清0。
6	I2CMBUSY	I ² C主机忙碌状态位。 当主机忙于处理事务时, 该位被置1。 当主机处于就绪状态或者当另一主机取得了总线控制权时, 该位被清0。
5	I2CAL	I ² C仲裁失效状态位。 当I ² C主机未获得对I ² C总线的控制权时, 该位被置1。 如果I2CxMCTL寄存器的I2CALENI位被置1, 当该位的值为1时, 使能中断。 在其它条件下, 该位被清0。
4	I2CMNA	I ² C主机NACK地址位。 当主机从一个I ² C从机地址接收到NACK条件时, 该位被置1。 如果I2CxMCTL寄存器的I2CNACKENI位被置1, 当该位的值为1时, 使能中断。 在其它条件下, 该位被清0。
3	I2CMRXQ	I ² C主机接收请求位。 当数据进入接收FIFO后, 该位被置1。如果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 ² C主机发送请求位。 如果发送FIFO为空或仅包含一个字节, 且主机已经传输一个地址并执行了写操作, 则该位置1。如果I2CxMCTL寄存器的I2CMTENI位被置1, 当该位的值为1时, 使能中断。 在其它条件下, 该位被清0。
1:0	I2CMTFSTA	I ² C主机发送FIFO状态位。 00 = I ² C主机发送FIFO为空。 01 = 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 ² C主机发送FIFO已满。

ADuC7122

I²C主机接收寄存器

名称: I2C0MRX, I2C1MRX
地址: 0xFFFF0888, 0xFFFF0908
默认值: 0x00, 0x00
访问类型: 只读
功能: 该8位寄存器是I²C主机接收寄存器。

I²C主机发送寄存器

名称: I2C0MTX, I2C1MTX
地址: 0xFFFF088C, 0xFFFF090C
默认值: 0x00, 0x00
访问类型: 读/写
功能: 该8位寄存器是I²C主机发送寄存器。

I²C主机读取计数寄存器

名称: I2C0MCNT0, I2C1MCNT0
地址: 0xFFFF0890, 0xFFFF0910
默认值: 0x0000, 0x0000
访问类型: 读/写
功能: 该16位寄存器用于保存主机需要从从机中读取的字节数。

I²C主机当前读取计数寄存器

名称: I2C0MCNT1, I2C1MCNT1
地址: 0xFFFF0894, 0xFFFF0914
默认值: 0x00, 0x00
访问类型: 读操作
功能: 该8位寄存器用于保存在一次读序列中主机从从机处接收到的字节数。

表103. I2CxMCNT0寄存器位功能描述(地址 = 0xFFFF0890、0xFFFF0910, 默认值 = 0x0000)

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	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, 则应将这些位清0。

I²C地址0寄存器

名称:	I2C0ADR0, I2C1ADR0
地址:	0xFFFF0898, 0xFFFF0918
默认值:	0x00, 0x00
访问类型:	读/写
功能:	当主机开始与从机进行通信后, 该8位寄存器用于保存7位从机地址和读/写位。

I²C地址1寄存器

名称:	I2C0ADR1, I2C1ADR1
地址:	0xFFFF089C, 0xFFFF091C
默认值:	0x00, 0x00
访问类型:	读/写
功能:	该8位寄存器只可用于10位寻址模式。它包含地址的最低有效字节。

I²C主机时钟控制寄存器

名称:	I2C0DIV, I2C1DIV
地址:	0xFFFF08A4, 0xFFFF0924
默认值:	0x1F1E, 0x1F1F
访问类型:	读/写
功能:	该寄存器用于控制SCL引脚上由主机产生的I ² C时钟的频率。详情见I2C部分。

I²C起始字节寄存器

名称:	I2C0SBYTE, I2C1SBYTE
地址:	0xFFFF08A0, 0xFFFF0920
默认值:	0x00, 0x00
访问类型:	读/写
功能:	此寄存器可用来在处理开始时产生一个起始字节。

要产生一个起始字节(后接正常地址), 首先应写入I2CxSBYTE, 然后写入地址寄存器(I2CxADR_x)。这将把写入I2CxSBYTE的字节驱动到总线上, 其后跟随一个重复起始条件。此寄存器可用来将任何字节驱动到I²C总线上, 然后是一个重复起始条件(不只是一个起始字节, 例如00000001)。

表104. 7位地址模式下的I2CxADR0寄存器(地址 = 0xFFFF0898、0xFFFF0918, 默认值 = 0x00)

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105. 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。

表106. 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]。

表107. I2CxDIV寄存器

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

ADuC7122

I²C从机寄存器

I²C从机控制寄存器

名称:	I2C0SCTL, I2C1SCTL
地址:	0xFFFF08A8, 0xFFFF0928
默认值:	0x0000, 0x000
访问类型:	读/写
功能:	该16位寄存器用于在从机模式下配置I ² C外设。

表108. 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 ² C停止条件检测中断使能位。 将该位置1, 则当从I ² C总线上检测到停止条件时, 使能中断。 清除该中断源。
7	I2CNACKEN	I ² C NACK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(NACK)。 将该位清0, 可允许硬件对ACK/NACK序列进行控制。
6		保留。应向此位写入0值。
5	I2CSETEN	I ² 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SCLx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SCLx负向沿之后立刻产生发送请求中断。
4	I2CGCLR	I ² 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(I2CGC)与ID (I2CGCID[1:0])位。 在其它状况下, 该位被清0。
3	I2CHGCEN	I ² 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 and I2CxALT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, 可使用该方法。该广播消息会发送到总线上的所有主机。ADuC7122会监视这些地址。要求主机注意的器件会将地址嵌入到消息中。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 ² 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 ² C广播使能位。 将该位置1, 可以让从机为I ² C广播发送有效应答, 写地址0x00。然后器件将识别一个数据位。如果器件接收到的数据是0x06, 即由硬件复位和对从机地址的可编程部分进行写操作, 那么, 根据2000年1月的I ² C总线规范, 此时I ² C接口复位。这个命令可用于复位整个I ² C系统。如果接收到的数据为0x04, 即由硬件对从机地址可编程部分进行写操作, 则当产生任一广播后, 广播中断状态位被置位。 在复位后, 用户必须通过重新对器件地址进行编程来进行恰当的操作。 通过将该位置1, 可允许从机应答I ² C广播命令。 通过将该位清0, 可禁用广播命令识别。

位	名称	说明
1	ADR10EN	I ² C 10位地址模式。 将该位置1，以启用10位地址模式。 将该位清0，以启用正常地址模式。
0	I2CSEN	I ² C从机使能位。 用户置1，使能I ² C从机模式。 清0，禁用I ² C从机模式。

I²C从机状态寄存器

名称： I2C0SSTA, I2C1SSTA
地址： 0xFFFFF08AC, 0xFFFFF092C
默认值： 0x0000, 0x0000
访问类型： 只读
功能： 该16位寄存器是从机模式下的I²C状态寄存器。

表109. I2CxSSTA寄存器位功能描述

位	名称	说明
15		保留位。
14	I2CSTA	检测到起始条件并且地址匹配时，该位置1。如果接收到起始字节(0x01)，或者使能了广播且接收到广播代码0x00，该位也会置1。 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 ² 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 ² 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 ² C广播ID位。 00 = 未收到广播。 01 = 广播复位和程序地址。 10 = 通用程序地址。 11 = 广播匹配可供选择的ID。 注意：无法通过广播复位命令来清除这些位。 通过向I2CxSCTL的I2CGCCLR位写入1，可清除这些位。
7	I2CGC	I ² C广播状态位。 从机接收到任何类型的广播命令后，该位置1。 从机接收到复位命令后，寄存器返回各自的默认状态。 从机接收到硬件广播命令后，接收FIFO将保存命令的第2个字节，它可以与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 ² C从机忙碌状态位。 从机接收到起始条件后，该位置1。 发生下列情形之一，硬件会自动清除该位：如接收到的地址与I2CxIDx寄存器的内容不匹配，从机收到停止条件，或重复起始地址与I2CxIDx寄存器的内容不匹配。
5	I2CSNA	I ² C从机NACK数据位。 从机发出非应答信号(NACK)以响应总线地址时，该位被置1。如果因发送FIFO内无数据而返回一非应答信号，或者I2CxSCTL寄存器的I2CNACKEN位被置1，则该位置1。 在其它条件下，该位被清0。

ADuC7122

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 ² C从机接收请求位。 从机接收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 ² C从机发送请求位。 从机接收到一个匹配的地址并执行读操作后, 该位被置1。 如果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, 此位立即被置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, 此位立即被置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 ² C从机FIFO下溢状态位。 主机要求从机发送数据时, 如果发送FIFO为空, 则该位变为高电平。在读位操作期间, 该位在SCL上升沿置位。 在其它条件下, 该位被清0。
0	I2CETSTA	I ² 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, 此位变为高电平。 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, 此位立即被置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。

I²C从机接收寄存器

名称: I2C0SRX, I2C1SRX
 地址: 0xFFFF08B0, 0xFFFF0930
 默认值: 0x00
 访问类型: 读操作
 功能: 该8位寄存器是I²C从机接收寄存器。

I²C从机发送寄存器

名称: I2C0STX, I2C1STX
 地址: 0xFFFF08B4, 0xFFFF0934
 默认值: 0x00
 访问类型: 读/写
 功能: 该8位寄存器是I²C从机发送寄存器。

I²C硬件广播识别寄存器

名称: I2C0ALT, I2C1ALT
 地址: 0xFFFF08B8, 0xFFFF0938
 默认值: 0x00
 访问类型: 读/写
 功能: 2CxSCTL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。

I²C从机ID寄存器

名称: I2C0IDx, I2C1IDx
 地址: 0xFFFF093C = I2C1ID0
 0xFFFF08BC = I2C0ID0
 0xFFFF0940 = I2C1ID1
 0xFFFF08C0 = I2C0ID1
 0xFFFF0944 = I2C1ID2
 0xFFFF08C4 = I2C0ID2
 0xFFFF0948 = I2C1ID3
 0xFFFF08C8 = I2C0ID3
 默认值: 0x00
 访问类型: 读/写
 功能: 通过编程, 可在这些8位寄存器内编程从机的I²C总线ID。详情见I²C总线地址部分。

ADuC7122

I²C公共寄存器

I²C FIFO状态寄存器

名称: I2C0FSTA, I2C1FSTA

地址: 0xFFFFF08CC, 0xFFFFF094C

默认值: 0x0000

访问类型: 读/写

功能: 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。

表110. I2CxFSTA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 ² C主机接收FIFO状态位。 00 = FIFO为空。 01 = FIFO字节写入。 10 = FIFO中有1字节数据。 11 = FIFO已满。
5:4	I2CMTXSTA	I ² C主机发送FIFO状态位。 00 = FIFO为空。 01 = FIFO字节写入。 10 = FIFO中有1字节数据。 11 = FIFO已满。
3:2	I2CSRSTA	I ² C从机接收FIFO状态位。 00 = FIFO为空。 01 = FIFO字节写入。 10 = FIFO中有1字节数据。 11 = FIFO已满。
1:0	I2CSTXSTA	I ² C从机发送FIFO状态位。 00 = FIFO为空。 01 = FIFO字节写入。 10 = FIFO中有1字节数据。 11 = FIFO已满。

串行外设接口

ADuC7122片内集成一个完整的硬件串行外设接口(SPI)。

SPI是一种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 Mb。

该SPI端口可配置为主机或从机操作，一般由4个引脚组成：SPIMISO、SPIMOSI、SPICLK和SPIC $\overline{\text{S}}$ 。

SPIMISO(主机输入，从机输出)引脚

SPIMISO引脚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SPIMISO线路(数据输入)应与从机内的SPIMISO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。

SPIMOSI(主机输出，从机输入)引脚

SPIMOSI引脚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SPIMOSI线路(数据输出)应与从机内的SPIMOSI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。在主机模式下，SPICLK引脚配置成输出；而在从机模式下，配置成输入。

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

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

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接收数据(速率可达10 Mb)。

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(SPIC $\overline{\text{S}}$ 输入)引脚

在SPI从机模式时，置位SPIC $\overline{\text{S}}$ 引脚将启动数据传输，该引脚为一个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SPIC $\overline{\text{S}}$ 解除置位以结束传输为止。在从机模式下，SPIC $\overline{\text{S}}$ 总是输入。

在SPI主机模式下，SPIC $\overline{\text{S}}$ 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

ADuC7122的SPI引脚为P0.2至P0.5。

P0.5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为SPICLK引脚。

P0.3为主机输入、从机输出(SPIMISO)引脚。

P0.4为主机输出、从机输入(SPIMOSI)引脚。

要配置P0.2至P0.5在SPI模式下工作，参见“通用I/O”部分。

ADuC7122

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

名称： SPISTA
 地址： 0xFFFF0A00
 默认值： 0x0000
 访问类型： 只读
 功能： 该16位寄存器用于存储主机、从机模式下SPI接口的状态。

表111. 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寄存器中的SPIRXMDE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 = 该FIFO内有1个有效字节。 010 = 该FIFO内有2个有效字节。 011 = 该FIFO内有3个有效字节。 100 = 该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 = 该FIFO内有1个有效字节。 010 = 该FIFO内有2个有效字节。 011 = 该FIFO内有3个有效字节。 100 = 该FIFO内有4个有效字节。 该位清0，可使能7位地址模式。
0	SPISTA	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

名称: SPIRX
 地址: 0xFFFF0A04
 默认值: 0x00
 访问类型: 读操作
 功能: 该8位寄存器是SPI接收寄存器。

SPITX寄存器

名称: SPITX
 地址: 0xFFFF0A08
 默认值: 0x00
 访问类型: 写操作
 功能: 该8位寄存器是SPI发送寄存器。

SPIDIV寄存器

名称: SPIDIV
 地址: 0xFFFF0A0C
 默认值: 0x1B
 访问类型: 读/写
 功能: 该8位寄存器是SPI波特率选择寄存器。

SPI控制寄存器

名称: SPICON
 地址: 0xFFFF0A10
 默认值: 0x0000
 访问类型: 读/写
 功能: 该16位寄存器用于在主机和从机模式下配置SPI外设。

表112. 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接收到三个或以上字节后, 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。清空使能位置1时, FIFO在单个微处理器周期内清空。 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。清空使能位置1时, FIFO在单个微处理器周期内清空。 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。 SPIC \overline{S} 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, SPIMISO在正常模式下工作。 将该位清0, 可禁用SPIMISO引脚上的输出驱动。该位被清0后, SPIMISO引脚变为开漏极。

ADuC7122

位	名称	说明
8	SPIROW	SPIRX上溢覆盖使能。 用户置1，则新接收到的串行字节覆盖接收寄存器中的有效数据。 用户清0，则新接收到的串行字节会被丢弃。
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	SPI TMDE	SPI传输和中断模式。 用户将该位置1，启动数据传输，同时向SPITX寄存器写入数据。只有当发送寄存器为空时，才产生中断。 用户清0，读取SPIRX寄存器开始发送。只有当接收寄存器已满时，才产生中断。
5	SPI LF	LSB优先传输使能位。 用户置1，先发送LSB。 用户清0，先发送MSB。
4	SPI WOM	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。 清0使用正常输出电平。
3	SPI CPO	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	SPI CPH	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	SPI MEN	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	SPI EN	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)

ADuC7122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35所示。

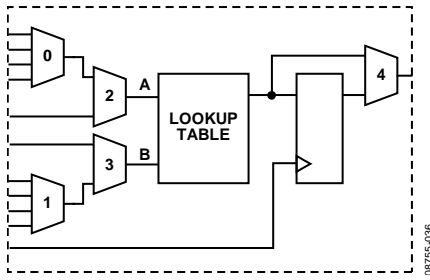


图35. PLA单元

ADuC7122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。

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CONVST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)的输出可以反馈到单元0(模块0)的多路复用器0的输入0
- 单元7(模块0)的输出可以反馈到单元8(模块1)的多路复用器0的输入0

表113. 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

PLA寄存器接口

PLA外设接口包括21个寄存器，具体描述参见表114至表128。

表114. PLAELMx寄存器

名称	地址	默认值	访问类型
PLAELM0	0xFFFF0B00	0x0000	R/W
PLAELM1	0xFFFF0B04	0x0000	R/W
PLAELM2	0xFFFF0B08	0x0000	R/W
PLAELM3	0xFFFF0B0C	0x0000	R/W
PLAELM4	0xFFFF0B10	0x0000	R/W
PLAELM5	0xFFFF0B14	0x0000	R/W
PLAELM6	0xFFFF0B18	0x0000	R/W
PLAELM7	0xFFFF0B1C	0x0000	R/W
PLAELM8	0xFFFF0B20	0x0000	R/W
PLAELM9	0xFFFF0B24	0x0000	R/W
PLAELM10	0xFFFF0B28	0x0000	R/W
PLAELM11	0xFFFF0B2C	0x0000	R/W
PLAELM12	0xFFFF0B30	0x0000	R/W
PLAELM13	0xFFFF0B34	0x0000	R/W
PLAELM14	0xFFFF0B38	0x0000	R/W
PLAELM15	0xFFFF0B3C	0x0000	R/W

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115和表118。

ADuC7122

表 115. 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118)。
8:7		Mux1控制位(见表118)。
6	1	Mux2控制位。 用户置1, 选择Mux0的输出。
	0	用户清0, 选择PLADIN寄存器中的位值。
5	1	Mux3控制位。 置1, 选择特定单元的输入引脚。
	0	清0, 选择Mux1的输出。
4:1		查找表控制位。
	0000	0。
	0001	或非。
	0010	B与A非。
	0011	A非。
	0100	A与B非。
	0101	B非。
	0110	异或。
	0111	与非。
	1000	与。
	1001	同或。
	1010	B。
1011	A非或B。	
1100	答:	
1101	A或B非。	
1110	或。	
1111	1。	
0		Mux4控制位。置1, 旁路触发器。 清0, 使用触发器(默认为清0)。

表116. PLACLK寄存器

名称	地址	默认值	访问类型
PLACLK	0xFFFF0B40	0x00	R/W

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

表117. 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引脚连接的GPIO时钟。
	001	P0.0引脚连接的GPIO时钟。
	010	P0.7引脚连接的GPIO时钟。
	011	HCLK。
	100	外部晶振(OCLK) (32.768 kHz)。
	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引脚连接的GPIO时钟。
	001	P0.0引脚连接的GPIO时钟。
	010	P0.7引脚连接的GPIO时钟。
	011	HCLK。
	100	外部晶振(OCLK) (32.768 kHz)。
	101	定时器1溢出。
	其他	保留。

表118. 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

表119. PLAIRQ寄存器

名称	地址	默认值	访问类型
PLAIRQ	0xFFFF0B44	0x00000000	R/W

PLAIRQ可以使能IRQ0和/或IRQ1和选择IRQ中断源。

表120. PLAIRQ寄存器位功能描述

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

表121. PLAADC寄存器

名称	地址	默认值	访问类型
PLAADC	0xFFFF0B48	0x00000000	R/W

PLAADC是配置PLA作为ADC起始转换信号来源的寄存器。

表122. PLAADC寄存器位功能描述

位	值	说明
31:5		保留。
4	1 0	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。

表123. PLADIN寄存器

名称	地址	默认值	访问类型
PLADIN	0xFFFF0B4C	0x00000000	R/W

表124. PLADIN寄存器位功能描述

位	说明
31:16	保留。
15:0	单元15至单元0的输入位。

PLADIN是一个PLA的数据输入寄存器。

表125. PLADOUT寄存器

名称	地址	默认值	访问类型
PLADOUT	0xFFFF0B50	0x00000000	R

PLADOUT是一个PLA的数据输出寄存器。这个寄存器是始终更新的。

表126. 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。

表127. PLACLK寄存器

名称	地址	默认值	访问类型
PLACLK	0xFFFF0B40	0x00	W

PLACLK是PLA锁定选项。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。

中断系统

ADuC7122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寄存器和FIQ寄存器中的控制位都代表相同的中断源，如表128所示。

ADuC7122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。

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

表128. IRQ/FIQ¹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	保留	保留
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	I2C0主机IRQ	I ² C主机中断源位
16	I2C0从机IRQ	I ² C从机中断源位
17	I2C1主机IRQ	I ² C主机中断源位
18	I2C1从机IRQ	I ² 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

¹ 适用于IRQEN、FIQEN、IRQCLR、FIQCLR、IRQSTA和FIQSTA寄存器。

IRQ

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。器件有4个专门用于IRQ的32位寄存器：IRQSIG、IRQEN、IRQCLR和IRQSTA。

IRQSIG

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

IRQSIG寄存器

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

IRQEN

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。

IRQEN寄存器

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

IRQCLR

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

IRQCLR寄存器

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

IRQSTA

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

IRQSTA寄存器

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

快速中断请求(FIQ)

快速中断请求(FIQ)是进入处理器FIQ模式的一个异常信号。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

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

FIQSIG寄存器

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

FIQEN

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: 0xFFFFF0108
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。

FIQCLR寄存器

名称: FIQCLR
地址: 0xFFFFF010C
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: 0xFFFFF0100
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129。该寄存器允许控制可编程源中断。

表129. SWICFG寄存器位功能描述

位	说明
31至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IRQ。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。

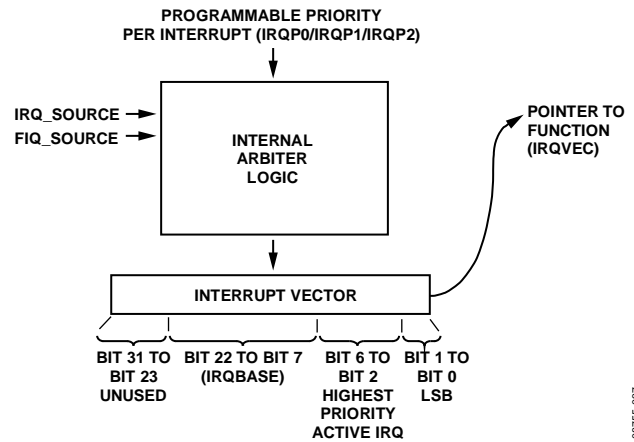


图36. 中断结构

矢量中断控制器(VIC)

ADuC7122集成一个增强的中断控制系统或矢量中断控制器。通过设置IRQCONN寄存器的位0，可以启用针对IRQ中断源的矢量中断控制器。同样，通过设置IRQCONN寄存器的位1，可以启用针对FIQ中断源的矢量中断控制器。矢量中断控制器在以下几个方面增强了标准IRQ/FIQ中断：

- 矢量中断——允许用户为每个中断源单独定义中断服务程序地址。这可以通过IRQBASE和IRQVEC寄存器来完成。
- IRQ/FIQ中断——可根据优先级进行嵌套，最多允许8级嵌套。FIQ中断的优先级高于IRQ中断。因此，当使能FIQ和IRQ的矢量中断控制器且优先级为最高时，将有可能形成16个不同的中断级。
- 可编程中断优先级——通过设置IRQP0至IRQP2寄存器，可以为中断源分配不同的中断优先级(1至8)。

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130. 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	R/W	0	矢量基地址

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。当产生IRQ中断, 且已通过设置IRQCONN的位0启用IRQ中断嵌套功能后, 该寄存器为只读的。

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

表131. 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	R/W	0	IRQBASE寄存器值。
6:2	只读	0	最高优先级IRQ中断源。这些位存储的数据对应0至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1, 那么这些位的值为00011。
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132. IRQP0寄存器位功能描述

位	名称	说明
31:27	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	TOPI	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	中断0不能设置优先级。

IRQP1寄存器

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

表133. IRQP1寄存器位功能描述

位	名称	说明
31	保留	保留位。
30:28	I2C0MPI	为I2C0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	ADCP1	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134. IRQP2寄存器位功能描述

位	名称	说明
31	保留	保留位。
30:28	PWMPI	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I2C1从机设置中断优先级(0至7)。
7	保留	保留位。
6:4	I2C1MPI	为I2C1主机设置中断优先级(0至7)。
3	保留	保留位。
2:0	I2C0SPI	为I2C0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

表135. 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PLA0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。此外，也无法设置中断源的中断优先级。在此默认状态下，FIQ的优先级高于IRQ的优先级。

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

表136. 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。例如，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。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137. 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。

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。当产生FIQ中断，且已通过设置IRQCONN的位1启用FIQ中断嵌套功能后，该寄存器为只读的。

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

表138. 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	R/W	0	IRQBASE寄存器值。
6:2		0	最高优先级FIQ中断源。这些位存储的数据对应0至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1，那么这些位的值为00011。
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。例如，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。

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

表139. 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)

ADuC7122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

ADuC7122

表140. 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。
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: 只写

表141. IRQCLRE寄存器位功能描述

位	名称	说明
31:26	保留	这些位为保留位, 不允许向其中写入数据。
25	IRQ5CLRI	必须在IRQ5中断服务程序中向该位写入1, 以清除边沿。
24	IRQ4CLRI	必须在IRQ4中断服务程序中向该位写入1, 以清除边沿。
23	保留	此位为保留位, 不允许向其中写入数据。
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

ADuC7122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。

事件选择特性支持根据定时器0和定时器1灵活地产生中断。T0CON和T1CON可用来配置中断源, 如表142所示。当定时器0或定时器1到期时, 就会根据T0CON和T1CON寄存器中的事件选择产生中断。

表142. 事件选择编号

事件选择(TxCON [16:12])	中断号	名称
00000	2	定时器0
00001	3	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	定时器4
00101	7	保留
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	模数转换器
01011	13	UART
01100	14	SPI
01101	15	I2C0主机
01110	16	I2C0从机
01111	17	I2C1主机
10000	18	I2C1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143。

表143. 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TOVAL0和TOVAL1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TOLD以载入到计数器内。当前计数器值可从TOVAL0读取。定时器0有一个事件捕获寄存器(TOCAP)，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TOCAP内，与此同时，定时器继续运行。此功能可用来确定事件断言，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TOCLRI写操作后，定时器0将从TOLD重载数值。

定时器0接口有6个寄存器，如表144所示。

表144. 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)。
TOCLRI	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。

表145. 定时器0值寄存器

名称	地址	默认值	访问类型
TOVAL0	0xFFFF0304	0x00,	R
TOVAL1	0xFFFF0308	0x00	R

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。

表146. 定时器0捕捉寄存器

名称	地址	默认值	访问类型
TOCAP	0xFFFF0314	0x00	R

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。

表147. 定时器0控制寄存器

名称	地址	默认值	访问类型
TOCON	0xFFFF030C	0x00	R/W

该17位寄存器用于设置定时器0的工作模式。

表148. 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(ES)范围是0至17。事件说明参见“定时器”部分。
11		保留。
10:9	00	时钟选择。 内部32 kHz振荡器。
	01	UCLK。
	10	外部32 kHz晶振。
	11	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	定时器0工作模式。 16位工作模式(默认)。
	1	48位工作模式。
3:0	0000	预分频器。 时钟源/1(默认)。
	0100	时钟源/16。
	1000	时钟源/256。
	1111	时钟源/32,768。

表149. 定时器0载入寄存器

名称	地址	默认值	访问类型
TOLD	0xFFFF0300	0x00	R/W

TOLD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。

表150. 定时器0清零寄存器

名称	地址	默认值	访问类型
TOCLRI	0xFFFF0310	0x00	W

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。

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。此功能可用于更精确地判断事件断言。

定时器1接口有5个寄存器，如表151所示。

表151. 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。
T1CLR	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。

注意，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T1ICLR写操作后，定时器1将从T1LD载入数值。

表152. 定时器1载入寄存器

名称	地址	默认值	访问类型
T1LD	0xFFFF0320	0x00000	R/W

T1LD是一个32位寄存器，用于保存载入计数器的32位值。

表153. 定时器1清零寄存器

名称	地址	默认值	访问类型
T1CLR	0xFFFF032C	0x00	W

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。

表154. 定时器1值寄存器

名称	地址	默认值	访问类型
T1VAL	0xFFFF0324	0x0000	R

T1VAL是一个32位寄存器，用于保存定时器1的当前值。

表155. 定时器1捕捉寄存器

名称	地址	默认值	访问类型
T1CAP	0xFFFF0330	0x00	R

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。

表156. 定时器1控制寄存器

名称	地址	默认值	访问类型
T1CON	0xFFFF0328	0x0000	R/W

该32位寄存器用于设置定时器1的工作模式。

ADuC7122

表157. 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	1 0	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17。事件说明参见“定时器”部分。
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。
8	1 0	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	1 0	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	1 0	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。 时钟源/1(默认)。 时钟源/16。 时钟源/256。 时钟源/32,768。

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4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(见表128)。

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T2ICLR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158所示。

表158. 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。

表159. 定时器2载入寄存器

名称	地址	默认值	访问类型
T2LD	0xFFFF0340	0x00000	R/W

T2LD是一个32位寄存器，用于保存载入计数器的32位值。

表160. 定时器2清零寄存器

名称	地址	默认值	访问类型
T2CLR1	0xFFFF034C	0x00	W

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。

表161. 定时器2值寄存器

名称	地址	默认值	访问类型
T2VAL	0xFFFF0344	0x0000	R

T2VAL为32位寄存器，用于保存定时器2的当前值。

表162. 定时器2控制寄存器

名称	地址	默认值	访问类型
T2CON	0xFFFF0348	0x0000	R/W

该32位寄存器用于设置定时器2的工作模式。

ADuC7122

表163. T2CON寄存器位功能描述

位	值	说明
31:11		保留。
10:9		时钟源选择。
	00	内部32.768 kHz振荡器(默认)。
	01	内核时钟。
	10	外部32.768 kHz时钟晶体。
	11	UCLK。
8		递增计数。
	1	用户置1, 定时器2递增计数。
	0	用户清0, 定时器2递减计数(默认)。
7		定时器2使能位。
	1	用户置1, 使能定时器2。
	0	用户清0, 禁用定时器2(默认)。
6		定时器2模式。
	1	用户置1, 选择周期模式。
	0	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	时钟源/1(默认)。
	0100	时钟源/16。
	1000	源时钟/256(此设置应与定时器2格式10和格式11一起使用)。
	1111	时钟源/32,768。

定时器3—看门狗定时器

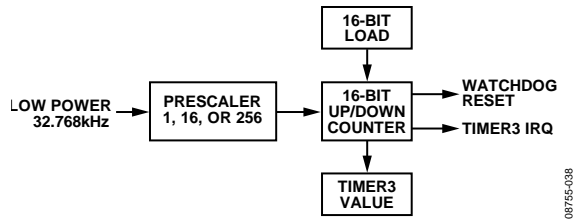


图37. 定时器3框图

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T3ICLR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，如下表所示。

表164. 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。
T3ICLR	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。

表165. 定时器3载入寄存器

名称	地址	默认值	访问类型
T3LD	0xFFFF0360	0x03D7	R/W

该16位寄存器用于保存定时器3的重新载入值。

表166. 定时器3值寄存器

名称	地址	默认值	访问类型
T3VAL	0xFFFF0364	0x03D7	R

该16位只读寄存器用于保存定时器3的当前计数值。

表167. 定时器3清零寄存器

名称	地址	默认值	访问类型
T3ICLR	0xFFFF036C	0x00	W

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。

表168. 定时器3控制寄存器

名称	地址	默认值	访问类型
T3CON	0xFFFF0368	0x0000	R/W一次

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169。

ADuC7122

表169. T3CON寄存器位功能描述

位	值	说明
16:9		这些位保留，应由用户代码写入0。
8	1 0	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	1 0	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	1 0	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	1 0	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	1 0	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。 时钟源/1(默认)。 保留。 保留。 保留。
1	1 0	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	1 0	PD_OFF。 用户代码置1，在利用POWCON寄存器的位[6:4]使外设掉电时停止定时器3。 用户代码清0，在利用POWCON寄存器的位[6:4]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(T3CPM[4])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(参见图38)。

在进入看门狗模式之前必须先向T3ICLR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,0x00不应该作为初始值种子,因为0x00会一直迫使系统快速复位。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。

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位数值格式或者时间格式(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。此功能用来更精确地判断事件断言。

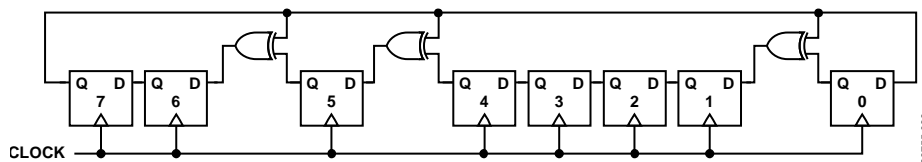


图38. 8位LFSR

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T4CLRI写操作后,定时器4将从T4LD重载数值。

表170. 定时器4载入寄存器

名称	地址	默认值	访问类型
T4LD	0xFFFF0380	0x00000	R/W

T4LD是一个32位寄存器,用于保存载入计数器的32位值。

表171. 定时器4清零寄存器

名称	地址	默认值	访问类型
T4CLRI	0xFFFF038C	0x00	W

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。

表172. 定时器4值寄存器

名称	地址	默认值	访问类型
T4VAL	0xFFFF0384	0x0000	R

T4VAL是一个32位寄存器,用于保存定时器4的当前值。

表173. 定时器4捕捉寄存器

名称	地址	默认值	访问类型
T4CAP	0xFFFF0390	0x00	R

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。

表174. 定时器4控制寄存器

名称	地址	默认值	访问类型
T4CON	0xFFFF0388	0x0000	R/W

该32位寄存器用于设置定时器4的工作模式。

ADuC7122

表175. T4CON寄存器位功能描述

位	值	说明
31:18		保留。用户置0。
17	1 0	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。
11:9	000 001 010 011	时钟选择。 32.768 kHz振荡器。 内核时钟。 UCLK。 UCLK。
8	1 0	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	1 0	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	1 0	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	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。 时钟源/1(默认)。 时钟源/16。 时钟源/256。 时钟源/32,768。

硬件设计考虑

电源

ADuC7122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39是一个典型的分离电源配置。

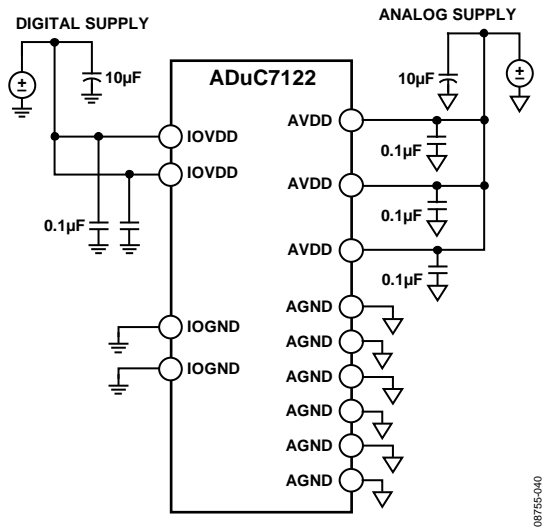


图39. 外部双电源连接

除了使用两个分离的电源外,用户还可以通过在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40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。

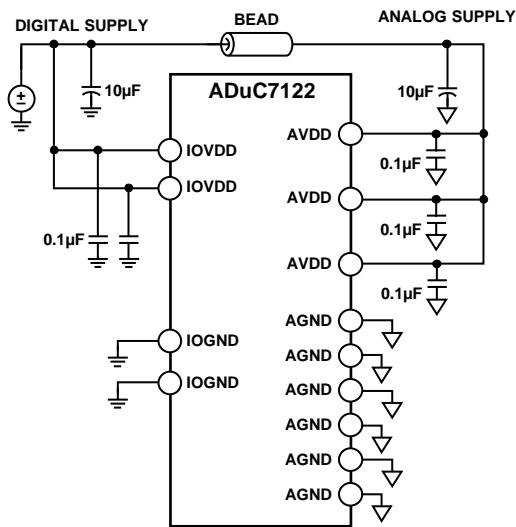


图40. 外部单电源连接

注意在图39和图40中,在IOVDD处有一个大容值(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,ADuC7122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41所推荐的滤波电路。

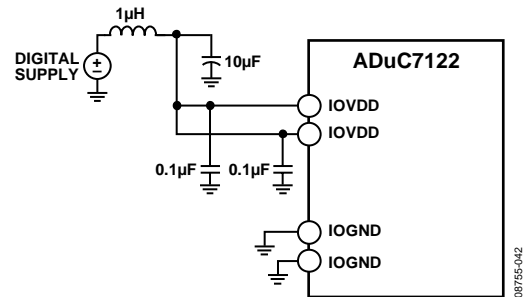


图41. 推荐的IOVDD电源滤波电路

ADuC7122

线性稳压器

ADuC7122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。LVDD引脚的2.6 V电源用来给内核逻辑供电。在各LVDD和DGND之间必须连接一个0.47 μF 的补偿电容(应尽量靠近这些引脚)作为电荷槽，如图42所示。内部LDO提供稳定的2.5 V电源。REG_PWR引脚提供2.5 V电源输出。在REG_PWR和DGND之间必须连接一个0.47 μF 的补偿电容(应尽量靠近这些引脚)作为电荷槽，如图42所示。

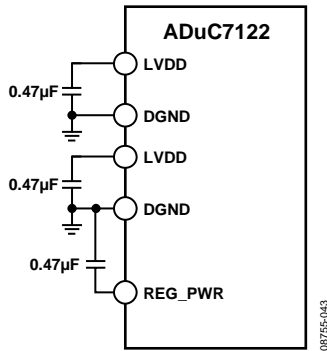


图42. 稳压器连接

LVDD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ADuC7122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43a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，ADuC7122的所有AGND和IOGND引脚都应连接到模拟地平面，如图43b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。ADuC7122可以放置在数字和模拟部分之间，如图43c所示。

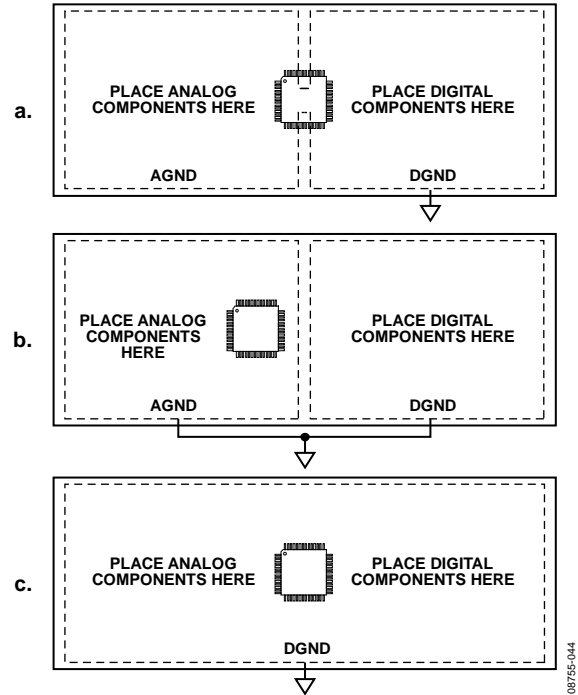


图43. 系统地框图

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43b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43c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，尽量少用一些或不要用迹线将引脚通过其过孔与地分离。

当ADuC7122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 Ω 或200 Ω 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

ADuC7122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44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。

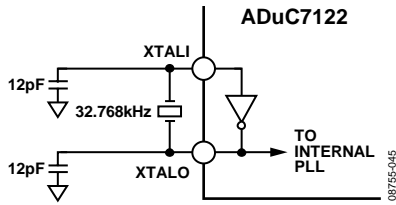


图44. 外部并行谐振晶体连接

如果使用一个外部时钟源输入代替锁相环(如图45所示)，PLLCON寄存器的位1和位0都需要修改，外部时钟从P1.4和XCLK引脚输入。

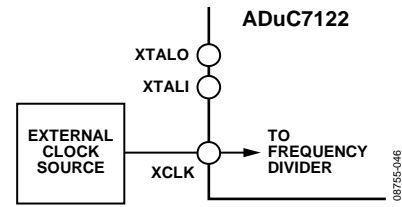
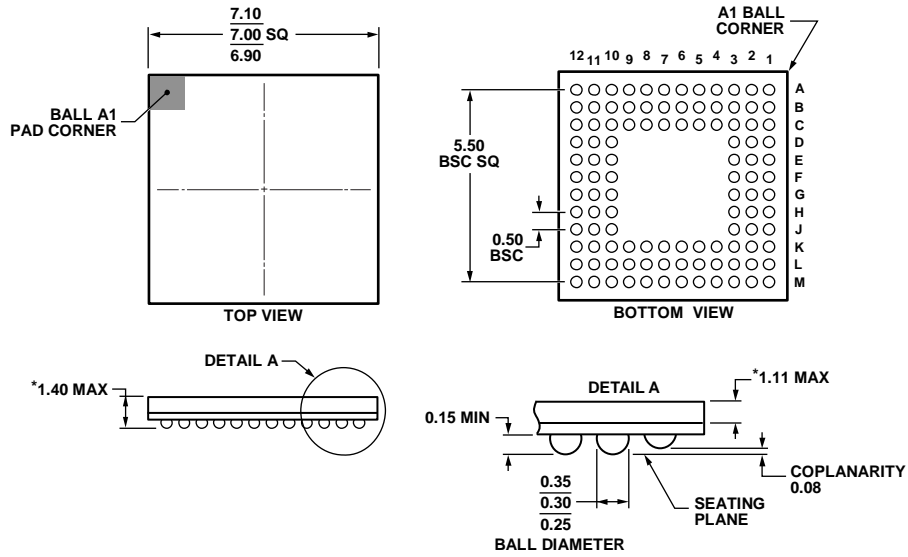


图45. 连接一个外部时钟源

使用外部时钟源时，ADuC7122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。

ADuC7122

外形尺寸



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

图46. 108引脚CSP_BGA封装
(BC-108-4)

尺寸单位: mm

000408-A

订购指南

型号 ¹	温度范围	封装描述	封装选项
ADuC7122BBCZ	-10°C至+95°C	108引脚CSP_BGA封装	BC-108-4
ADuC7122BBCZ-RL	-10°C至+95°C	108引脚CSP_BGA封装, 13"卷带和卷盘	BC-108-4
EVAL-ADUC7122QSPZ		ADuC7122 Quickstart Plus开发系统	

¹ Z = 符合RoHS标准的器件。

I²C指最初由Philips Semiconductors(现为NXP Semiconductors)开发的一种通信协议。