

ADuC7124/ADuC7126

特性

模拟输入/输出

多通道、12位、1 MSPS ADC

多达16个ADC通道

全差分模式和单端模式

模拟输入范围: 0 V至 V_{REF}

12位电压输出DAC

提供4个DAC输出

片内基准电压

片内温度传感器($\pm 3^\circ\text{C}$)

电压比较器

微控制器

16位/32位RISC架构ARM7TDMI内核

JTAG端口支持代码下载和调试

时钟选项

修正的片内振荡器($\pm 3\%$)

外部时钟晶体

可达41.78 MHz的外部时钟源

具有可编程分频器的41.78 MHz锁相环

存储器

126 kB Flash/EE存储器, 32 kB SRAM

在线下载, 基于JTAG调试

软件触发在线重新编程能力

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

片内外设

2个完全I²C兼容通道

SPI(主模式下20 Mbps, 从模式下10 Mbps)

输入级和输出级具有4字节FIFO

2个UART通道

输入级和输出级具有16字节FIFO

多达40个GPIO端口

所有GPIO均兼容5 V电压

4个通用定时器

看门狗定时器(WDT)和唤醒定时器

可编程逻辑阵列(PLA)

16个PLA元件

16位、6通道PWM

电源监控器

电源

额定工作电压: 3 V

主动模式: 11.6 mA(5 MHz), 33.3 mA(41.78 MHz)

封装和温度范围

额定工作温度范围: -40°C 至 $+125^\circ\text{C}$

64引脚LFCSP和80引脚LQFP

工具

低成本QuickStart开发系统

完全第三方支持

应用

工业控制和自动化系统

智能传感器、精密仪器

基站系统、光纤网络

病人监护

功能框图

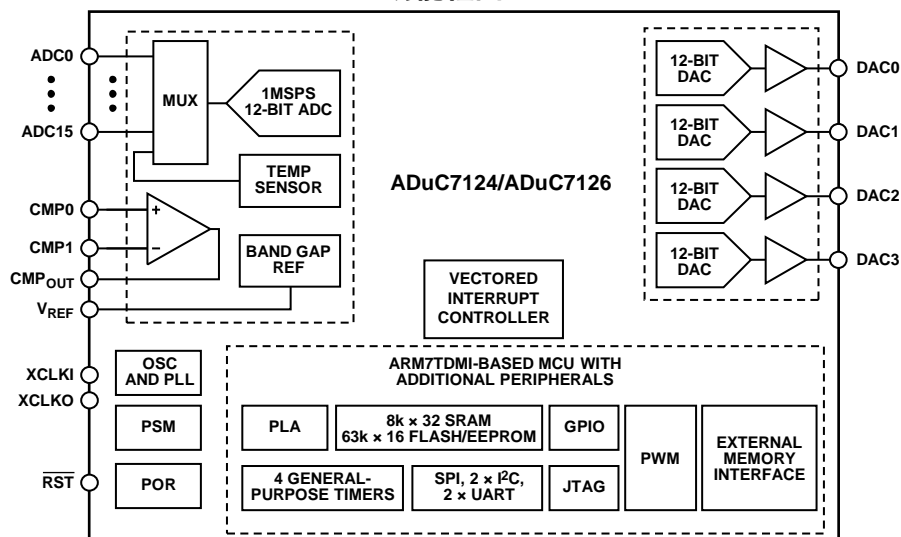


图1.

Rev. C

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.

ADI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, ADI不对翻译中存在的差异或由此产生的错误负责。如需确认任何词语的准确性, 请参考ADI提供的最新英文版数据手册。

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.

Tel: 781.329.4700

www.analog.com

Fax: 781.461.3113 ©2010–2012 Analog Devices, Inc. All rights reserved.

目录

特性.....	1	带隙基准电压源.....	43
应用.....	1	非易失性Flash/EE存储器.....	44
功能框图.....	1	编程.....	44
修订历史.....	3	Flash/EE存储器安全性.....	45
概述.....	4	Flash/EE控制接口.....	45
技术规格.....	5	SRAM和Flash/EE执行时间.....	48
时序规格.....	8	复位和重映射.....	48
绝对最大额定值.....	13	其他模拟外设.....	51
ESD警告.....	13	DAC.....	51
引脚配置和功能描述.....	14	电源监控器.....	53
典型工作特性.....	23	比较器.....	53
术语.....	26	振荡器和锁相环—电源控制.....	54
ADC技术规格.....	26	数字外设.....	58
DAC技术规格.....	26	通用输入/输出.....	58
ARM7TDMI内核概览.....	27	串口多路复用器.....	60
Thumb模式(T).....	27	UART串行接口.....	60
长乘(M).....	27	串行外设接口.....	66
EmbeddedICE (I).....	27	I ² C.....	70
异常.....	27	PWM概述.....	78
ARM寄存器.....	27	可编程逻辑阵列(PLA).....	81
中断延迟.....	28	处理器相关外设.....	84
存储器结构.....	29	中断系统.....	84
存储器访问.....	29	IRQ.....	84
Flash/EE存储器.....	29	快速中断请求(FIQ).....	85
SRAM.....	29	矢量中断控制器(VIC).....	86
存储器映射寄存器.....	29	定时器.....	91
ADC电路概览.....	37	外部存储器接口.....	97
传递函数.....	37	硬件设计考虑.....	101
典型操作.....	38	电源.....	101
寄存器接口.....	38	接地和电路板布局建议.....	102
转换器操作.....	40	时钟振荡器.....	102
驱动模拟输入.....	41	上电复位操作.....	103
校准.....	42	外形尺寸.....	104
温度传感器.....	42	订购指南.....	105

修订历史**2012年5月—修订版B至修订版C**

将概述部分中的位变为字节	4
更改表2和表3	8
更改表4、图2和图3	9
更改表5和图4	10
更改表6和图5	11
更改表7和图6	12
更改表9中的引脚50和引脚51	14
更改“串行下载(在线编程)”部分	44
更改表77	57
更改表78	58
更改表90	60
更改“正常450 UART波特率生成”部分	61
更改“串行外设接口”部分	66
定时器部分增加公式，并增加“时：分：秒1/128格式”部分	91
更改图69	103
更新“外形尺寸”部分	104
更改“订购指南”部分	105

2011年1月—修订版A至修订版B

更改表1	5
------	---

2010年10月—修订版0至修订版A

增加ADuC7126	通篇
更改特性部分	1
移动图1	1
更改图1	1
更改概述部分	4
更改表1的下列参数：25°C时的电压输出、电压TC、主动模式下的IOV _{DD} 电流、暂停模式下的IOV _{DD} 电流	5
更改表8	13
REFGND更改为GND _{REF}	13
更改图7和表9	14
增加图8和表10；重新排序	18
更改图17的标题	25
更改“存储器映射寄存器”部分	29
更改图26	30
更改表18	32
更改表21	33
更改表22	34
移动表25	35
更改表25	35
增加表26	35
更改表27	36
更改“温度传感器”部分	42
删除表59；重新排序	43

增加“通过I ² C执行下载(在线编程)”部分	44
更改“JTAG访问”部分和表37	45
更改表45	46
更改“RSTCFG寄存器”部分	49
删除表72和表75	49
删除表78	50
更改DAC部分、表62和表64	51
更改“ADC和DAC的基准电压”部分、表66、“在运算放大器模式下配置DAC缓冲器”部分、“DACBCFG寄存器”部分和表67	52
增加“DACBKEY1寄存器”部分和“DACBKEY2寄存器”部分	53
更改表69和图45	54
更改“外部晶体选择”部分和“外部时钟选择”部分	55
更改“PLLCON寄存器”和“POWCON0寄存器”部分	56
更改表78	58
更改表81	59
更改表84和表90	60
更改表93、“COM0FCR寄存器”部分、“COM1FCR寄存器”部分和表94	63
更改“串行外设接口”部分	66
更改“SPI寄存器”部分	67
更改“SPIDIV寄存器”部分和表101	68
更改“I ² C主机发送寄存器”部分	73
更改表109	74
更改“I ² C从机状态寄存器”部分	75
更改表113	79
更改表114标题和图50	80
更改“IRQCLRE寄存器”部分	90
更改图54	92
更改表141、“T1CLRI寄存器”部分和“T1CAP寄存器”部分	93
更改表143	94
增加“外部存储器接口”部分、表145、表146和图57	96
增加“XMCFG寄存器”部分、表147、表148、表149和表150	97
增加图58和图59	98
增加图60和图61	99
更改图62至图65	100
更改图67和图68	101
更改“上电复位操作”部分和图69	102
增加图71	103
更改“订购指南”部分	104

2010年9月—修订版0：初始版

概述

ADuC7124/ADuC7126均为完全集成的1 MSPS、12位数据采集系统，在单芯片内集成高性能多通道ADC、16位/32位MCU和Flash/EE存储器。

ADC具有多达12路单端输入。另外还有4个ADC输入通道也可以和4个DAC的输出引脚复用。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VREF。低漂移带隙基准电压源、温度传感器和电压比较器完善了ADC的外设置。

通过编程可以将DAC输出范围设置为三种电压范围之一。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。

这些器件通过一个片内振荡器和锁相环(PLL)产生41.78MHz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32 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。

ADuC7124/ADuC7126内置一个高级中断控制器。该矢量中断控制器(VIC)可以为每个中断分配一个优先级。它还支持嵌套中断，每个IRQ和FIQ最多允许8级嵌套。如果将IRQ和FIQ中断源合并，则可以支持总计16级嵌套中断。

片内出厂固件支持通过UART串行接口端口或I²C端口进行在线下载，并且支持通过JTAG接口进行非介入仿真。这些特性都集成在支持此MicroConverter[®]系列的低成本Quick-Start[™]开发系统中。

这些器件内置一个提供6路输出信号的16位PWM。

为便于通信，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。此外还有2个UART通道。每个UART含有一个可配置的16字节FIFO及接收和发送缓冲器。

这些器件的工作电压范围为2.7 V至3.6 V，额定温度范围为-40°C至+125°C工业温度范围。工作频率为41.78 MHz时，其典型功耗为120 mW。ADuC7124采用64引脚LFCSP封装。ADuC7126采用80引脚LQFP封装。

技术规格

除非另有说明， $AV_{DD} = IOV_{DD} = 2.7\text{ V}$ 至 3.6 V ， $V_{REF} = 2.5\text{ V}$ 内部基准电压， $f_{CORE} = 41.78\text{ MHz}$ ， $T_A = -40^\circ\text{C}$ 至 $+125^\circ\text{C}$ 。

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}					
分辨率	12			位	
积分非线性		± 0.6	± 1.5	LSB	2.5 V内部基准电压
		± 1.0		LSB	1.0 V外部基准电压
微分非线性 ^{3,4}		± 0.5	$+1/-0.9$	LSB	2.5 V内部基准电压
		$+0.7/-0.6$		LSB	1.0 V外部基准电压
直流代码分布		1		LSB	ADC输入为直流电压
端点误差 ⁵					
失调误差		± 1	± 2	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， $f_{SAMPLE} = 1\text{ MSPS}$ 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		-90		dB	在邻道上测量；未采样的输入通道 连接25 kHz正弦波信号
模拟输入					
输入电压范围 ⁴					
差模			$V_{CM}^6 \pm V_{REF}/2$	V	
单端模式			0 至 V_{REF}	V	
漏电流		± 1	± 6	μA	
输入电容		24		pF	在ADC采样期间
片内基准电压					在 V_{REF} 和AGND之间连接 $0.47\text{ }\mu\text{F}$ 电容
输出电压		2.5		V	
精度			± 5	mV	$T_A = 25^\circ\text{C}$
基准源温度系数		± 15		ppm/ $^\circ\text{C}$	
电源抑制比		80		dB	
输出阻抗		45		Ω	$T_A = 25^\circ\text{C}$
内部 V_{REF} 上电时间		1		ms	
外部基准输入					
输入电压范围	0.625		AV_{DD}	V	
DAC通道规格					$R_L = 5\text{ k}\Omega$ ， $C_L = 100\text{ pF}$
直流精度 ⁷					
分辨率		12		位	
相对精度		± 2		LSB	
微分非线性			± 1	LSB	保证单调性
失调误差			10	mV	2.5 V内部基准电压
增益误差 ⁸			1.0	%	
增益误差失配		0.1		%	DAC0满量程的百分比

ADuC7124/ADuC7126

参数	最小值	典型值	最大值	单位	测试条件/注释	
模拟输出						
输出电压范围0		0 至 DAC_{REF}		V	DAC_{REF} 范围: DAC_{GND} 至 $DAC_{V_{DD}}$	
输出电压范围1		0 至 2.5		V		
输出电压范围2		0 至 $DAC_{V_{DD}}$		V		
输出阻抗		0.5		Ω		
运算放大器模式下DAC						
运算放大器模式下DAC输出缓冲器						
输入失调电压		± 0.4		mV	5 k Ω 负载 $R_L = 5$ k Ω , $C_L = 100$ pF	
输入失调电压漂移		4		$\mu V/^\circ C$		
输入失调电流		2		nA		
输入偏置电流		2.5		nA		
增益		70		dB		
单位增益频率		4.5		MHz		
CMRR		78		dB		
建立时间		12		μs		
输出压摆率		3.2		V/ μs		
电源抑制比(PSRR)		75		dB		
DAC交流特性						
输出电压建立时间		10		μs		主进位1 LSB变化($DACxDAT$ 寄存器中同时变化的最大位数)
数模转换脉冲干扰		± 10		nV-sec		
比较器						
输入失调电压		± 15		mV	迟滞可以通过CMPCON寄存器中的CMPHYST位打开或关断 100 mV过驱、CMPRES = 11	
输入偏置电流		1		μA		
输入电压范围	AGND		$AV_{DD} - 1.2$	V		
输入电容		8.5		pF		
迟滞 ^{4,6}	2		15	mV		
响应时间		4		μs		
温度传感器						
25°C时电压输出		1.415		V	ADuC7124	
电压温度系数		1.392		V	ADuC7126	
精度		3.914		mV/ $^\circ C$	ADuC7124	
θ_{JA} 热阻		4.52		mV/ $^\circ C$	ADuC7126	
64引脚LFCSP		± 3		$^\circ C$	需要单点校准	
电源监控器(PSM)						
IOV _{DD} 跳变点选择		2.79		V	两个可选择跳变点	
电源跳变点精度		3.07		V		
上电复位		± 2.5		%	已选跳变点标称电压	
看门狗定时器(WDT)						
超时时间	0		512	秒		
FLASH/EE存储器						
耐久性 ⁹	10,000			周期	$T_j = 85^\circ C$	
数据保持 ¹⁰	20			年		
数字输入						
逻辑1输入电流		± 0.2	± 1	μA	除XCLKI和XCLKO外的所有数字输入 $V_{IH} = V_{DD}$ 或 $V_{IH} = 5$ V $V_{IL} = 0$ V; TDI、TDO和RTCK除外 $V_{IL} = 0$ V; TDI、TDO和RTCK	
逻辑0输入电流		-40	-60	μA		
输入电容		-80	-120	μA		
输入电容		5		pF		

参数	最小值	典型值	最大值	单位	测试条件/注释
逻辑输入 ³					除XCLKI外的所有数字输入
输入低电压 V_{INL}			0.8	V	
输入高电压 V_{INH}	2.0			V	
逻辑输出					除XCLKO外的所有数字输出
输出高电压 V_{OH}	2.4			V	$I_{SOURCE} = 1.6 \text{ mA}$
输出低电压 V_{OL} ¹¹			0.4	V	$I_{SINK} = 1.6 \text{ mA}$
晶体输入XCLKI和XCLKO 逻辑输入, 仅限XCLKI					
输入低电压 V_{INL}		0.8		V	
输入高电压 V_{INH}		1.6		V	
XCLKI输入电容		20		pF	
XCLKO输入电容		20		pF	
内部振荡器		32.768		kHz	
			±3	%	
MCU时钟速率 ⁴					
采用32 kHz内部振荡器		326		kHz	CD = 7
采用32 kHz外部晶体		41.78		MHz	CD = 0
使用外部时钟	0.05		44	MHz	$T_A = 85^\circ\text{C}$
	0.05		41.78	MHz	$T_A = 125^\circ\text{C}$
启动时间					内核时钟= 41.78 MHz
上电时		66		ms	
从暂停/休眠模式		2.6		μs	CD = 0
		247		μs	CD = 7
从休眠模式		1.58		ms	
从停止模式		1.7		ms	
可编程逻辑阵列(PLA)					
引脚传输延迟		12		ns	从输入引脚到输出引脚
单元传输延迟		2.5		ns	
电源要求 ^{12, 13}					
电源电压范围					
AV_{DD} 至AGND和 IOV_{DD} 至IOGND	2.7		3.6	V	
模拟电源电流					
AV_{DD} 电流		165		μA	ADC处于空闲模式
DAC V_{DD} 电流 ¹⁴		0.02		μA	
数字电源电流					
IOV_{DD} 主动模式下电流		8.1	12.5	mA	从Flash/EE执行代码
		11.6	17	mA	CD = 7
		33.3	50	mA	CD = 3
IOV_{DD} 暂停模式下电流		20.6	30	mA	CD = 0(时钟频率41.78 MHz)
IOV_{DD} 休眠模式下电流		110		μA	CD = 0(时钟频率41.78 MHz)
		600	680	μA	$T_A = 85^\circ\text{C}$
				μA	$T_A = 125^\circ\text{C}$
附加电源电流					
ADC		1.26		mA	1 MSPS时
		0.7		mA	62.5 kSPS时
DAC		315		μA	每DAC

ADuC7124/ADuC7126

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			3	kV	
最大FICDM通过电压			1.5	kV	

¹ 在内核正常工作时, 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。

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37所示)。当使用外部ADC系统元件时, 用户需要进行系统校准来消除外部端点误差并满足规格要求(详见校准部分)。

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 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。

⁸ DAC增益误差是使用一个递减的数据范围(100到内部2.5V基准电压)计算出来的。

⁹ 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。

¹⁰ 根据相当于 85°C 结温时的寿命。保持期限会随着结温递减。

¹¹ 测试是在最多8个I/O端口输出低电平时进行的。

¹² 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。

¹³ 在一个Flash/EE擦写周期中, I_{OVD} 电源电流通常提高2 mA。

¹⁴ 必须将此电流增加到 AV_{DD} 电流。

时序规格

I²C时序

表2. 快速模式下I²C时序(400 kHz)

参数	描述	从机		主机	单位
		最小值	最大值	典型值	
t_L	SCL低电平脉宽	200		1360	ns
t_H	SCL高电平脉宽	100		1140	ns
t_{SHD}	起始条件保持时间	300			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			μs
t_R	SCL和SDA的上升时间		300	200	ns
t_F	SCL和SDA的下降时间		300		ns

表3. 标准模式下I²C时序(100 kHz)

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

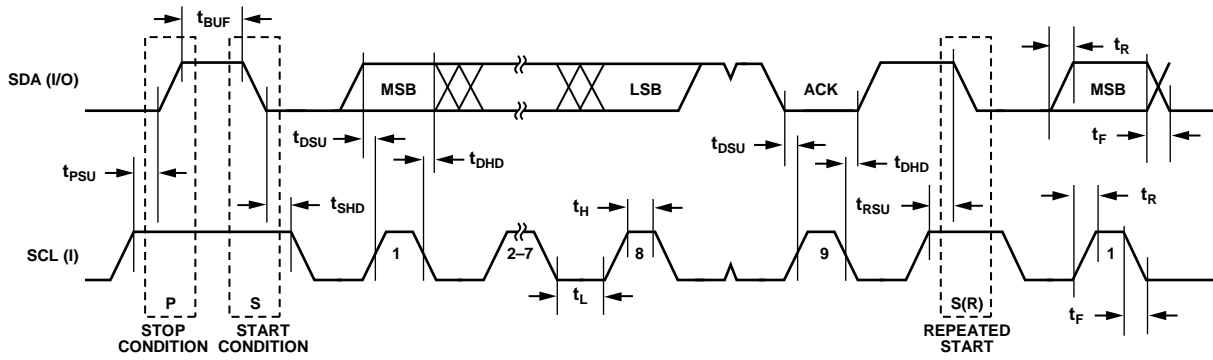


图2. I²C兼容接口时序

09123-029

SPI时序

表4. SPI主机定时(相位模式 = 1)

参数	描述	最小值	典型值	最大值	单位
t_{SL}	SCLK低电平脉宽 ¹		$(SPIDIV + 1) \times t_{UCLK}$		ns
t_{SH}	SCLK高电平脉宽 ¹		$(SPIDIV + 1) \times t_{UCLK}$		ns
t_{DAV}	SCLK边沿之后数据输出有效时间			25	ns
t_{DSU}	SCLK边沿之前数据输入建立时间 ¹	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK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}	SCLK上升时间		5	12.5	ns
t_{SF}	SCLK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。

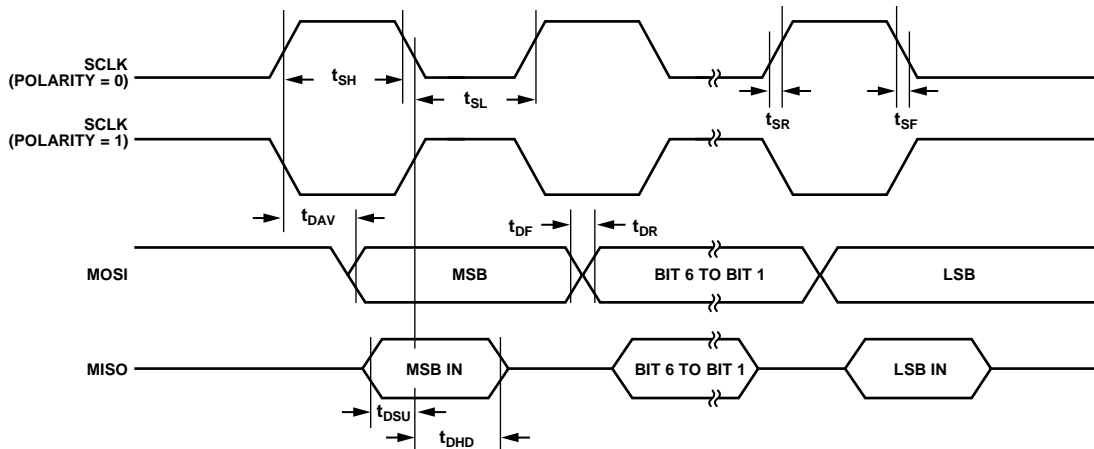


图3. SPI主机定时(相位模式 = 1)

09123-030

ADuC7124/ADuC7126

表5. SPI主机定时(相位模式 = 0)

参数	描述	最小值	典型值	最大值	单位
t_{SL}	SCLK低电平脉宽 ¹		$(SPIDIV + 1) \times t_{UCLK}$		ns
t_{SH}	SCLK高电平脉宽 ¹		$(SPIDIV + 1) \times t_{UCLK}$		ns
t_{DAV}	SCLK边沿之后数据输出有效时间			25	ns
t_{DOSU}	SCLK边沿之前数据输出建立时间			75	ns
t_{DSU}	SCLK边沿之前数据输入建立时间 ¹	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK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}	SCLK上升时间		5	12.5	ns
t_{SF}	SCLK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。

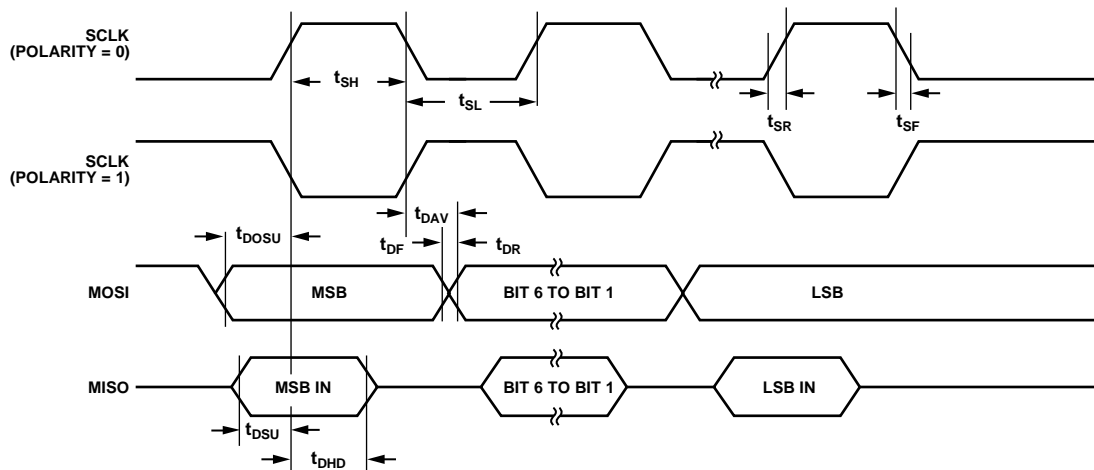


图4. SPI主机定时(相位模式 = 0)

09123-031

表6. SPI从机定时(相位模式 = 1)

参数	描述	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	\overline{CS} 至 SCLK 边沿	200			ns
t_{SL}	SCLK 低电平脉宽		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK 高电平脉宽		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{DAV}	SCLK 边沿之后数据输出有效时间			25	ns
t_{DSU}	SCLK 边沿之前数据输出建立时间 ¹	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK 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}	SCLK 上升时间		5	12.5	ns
t_{SF}	SCLK 下降时间		5	12.5	ns
t_{SFS}	\overline{CS} 在 SCLK 沿后变高	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。

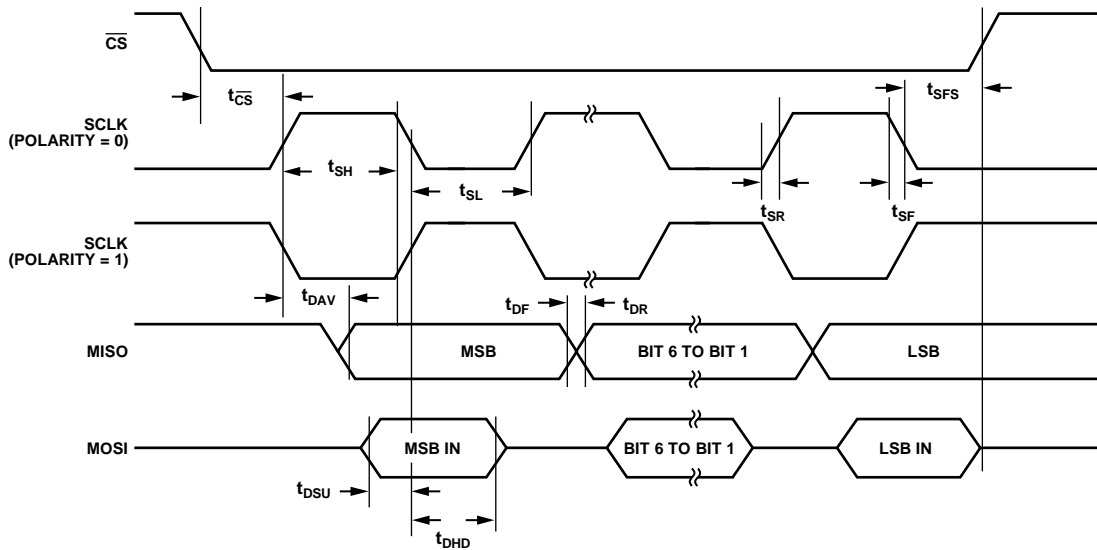


图5. SPI从机定时(相位模式 = 1)

09123-132

ADuC7124/ADuC7126

表7. SPI从机定时(相位模式 = 0)

参数	描述	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	\overline{CS} 至SCLK边沿	200			ns
t_{SL}	SCLK低电平脉宽		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK高电平脉宽		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{DAV}	SCLK边沿之后数据输出有效时间			25	ns
t_{DSU}	SCLK边沿之前数据输出建立时间 ¹	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK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}	SCLK上升时间		5	12.5	ns
t_{SF}	SCLK下降时间		5	12.5	ns
t_{DOCS}	\overline{CS} 边沿之后数据输出有效时间			25	ns
t_{SFS}	\overline{CS} 在SCLK沿后变高	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。

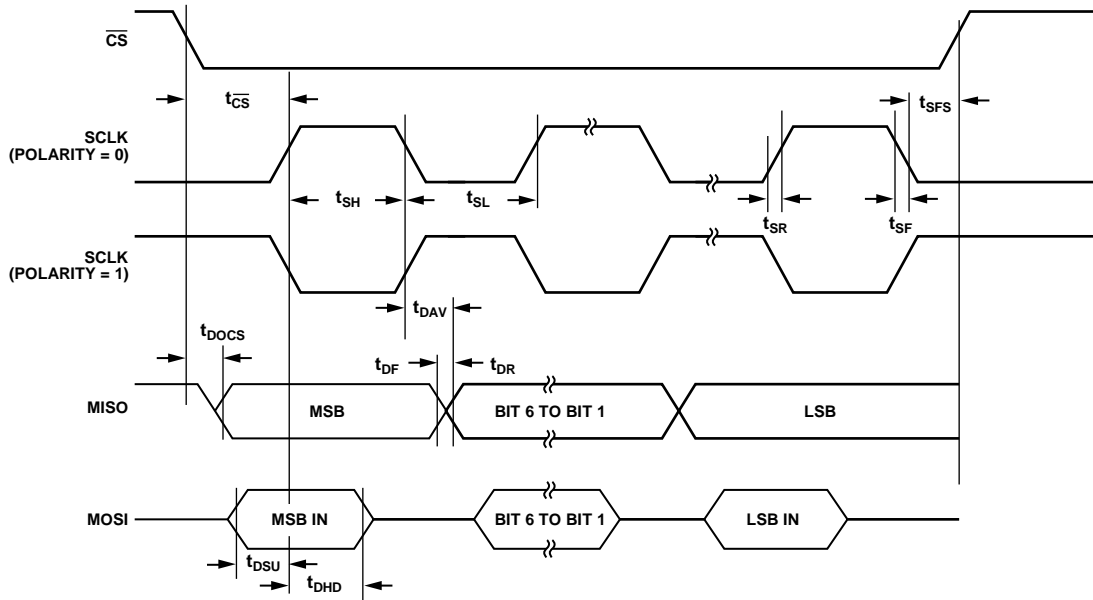


图6. SPI从机定时(相位模式 = 0)

09123-033

绝对最大额定值

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

$T_A = 25^\circ\text{C}$ 。

表8.

参数	额定值
AV_{DD} 至 IOV_{DD}	-0.3 V 至 +0.3 V
AGND 至 DGND	-0.3 V 至 +0.3 V
IOV_{DD} 至 IOGND, AV_{DD} 至 AGND	-0.3 V 至 +6 V
数字输入电压至IOGND	-0.3 V 至 +5.3 V
数字输出电压至IOGND	-0.3 V 至 $IOV_{DD} + 0.3\text{ V}$
V_{REF} 至 AGND	-0.3 V 至 $AV_{DD} + 0.3\text{ V}$
模拟输入至AGND	-0.3 V 至 $AV_{DD} + 0.3\text{ V}$
模拟输入至AGND	-0.3 V 至 $AV_{DD} + 0.3\text{ V}$
工业温度范围	-40°C 至 $+125^\circ\text{C}$
存储温度范围	-65°C 至 $+150^\circ\text{C}$
结温	150°C
θ_{JA} 热阻	
64引脚LFCSP	24°C/W
80引脚LQFP封装	38°C/W
回流焊峰值温度	
锡铅体系(10秒至30秒)	240°C
RoHS体系(20秒至40秒)	260°C

注意，超出上述绝对最大额定值可能会导致器件永久性损坏。这只是额定最值，并不能以这些条件或者在任何其它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

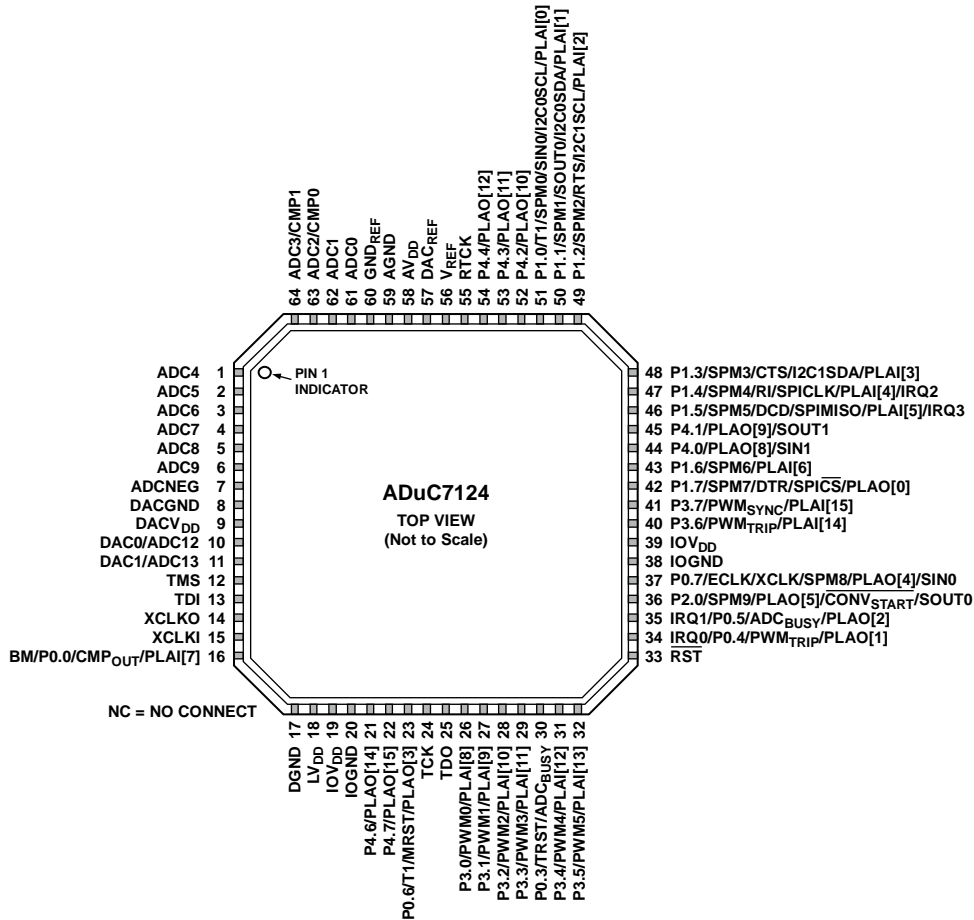


ESD(静电放电)敏感器件。

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。

ADuC7124/ADuC7126

引脚配置和功能描述



- NOTES**
1. THE EXPOSED PADDLE MUST BE SOLDERED TO THE PCB TO ENSURE PROPER HEAT DISSIPATION, NOISE, AND MECHANICAL STRENGTH BENEFITS.

图7. ADuC7124引脚配置

09123-107

表9. 引脚功能描述(ADuC7124 64引脚LFCSP)

引脚编号	引脚名称	描述
0	裸露焊盘	裸露焊盘。LFCSP_VQ有一个必须保持悬空的底部焊盘。
1	ADC4	单端或差分模拟输入4。
2	ADC5	单端或差分模拟输入5。
3	ADC6	单端或差分模拟输入6。
4	ADC7	单端或差分模拟输入7。
5	ADC8	单端或差分模拟输入8。
6	ADC9	单端或差分模拟输入9。
7	ADCNEG	伪差分模式下ADC偏置点或反相模拟输入。必须连接到要转换信号的地。该偏置点必须在0 V至1 V之间。
8	DACGND	DAC地。通常连接到AGND。
9	DACV _{DD}	DAC 3 V电压源。必须连接到AV _{DD} 。
10	DAC0/ADC12	DAC0电压输出(DAC0)。单端或差分模拟输入12 (ADC12)。
11	DAC1/ADC13	DAC1电压输出(DAC1)。单端或差分模拟输入13 (ADC13)。
12	TMS	JTAG测试端口输入，测试模式选择。调试和下载访问。
13	TDI	JTAG测试端口输入，测试数据输入。

引脚编号	引脚名称	描述
14	XCLKO	晶体振荡反相器输出。
15	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。
16	BM/P0.0/CMP _{OUT} /PLAI[7]	多功能输入输出引脚。 引导模式(BM)。复位时, 如果BM为低电平, 则ADuC7124进入下载模式; 如果BM通过1 kΩ电阻拉高, 则ADuC7124执行代码。 通用输入和输出端口0.0 (P0.0)。 电压比较器输出(CMP _{OUT})。 可编程逻辑阵列输入单元7 (PLAI[7])。
17	DGND	内核逻辑地。
18	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47μF电容器连接至DGND。
19	IOV _{DD}	3.3 V电源, 用于GPIO和片内稳压器输入。
20	IOGND	GPIO地。通常连接到DGND。
21	P4.6/PLAO[14]	通用输入和输出端口4.6 (P4.6)。 可编程逻辑阵列输出单元14 (PLAO[14])。
22	P4.7/PLAO[15]	通用输入和输出端口4.7 (P4.7)。 可编程逻辑阵列输出单元15 (PLAO[15])。
23	P0.6/T1/MRST/PLAO[3]	多功能引脚, 复位后输出低电平。 通用输出端口0.6 (P0.6)。 定时器1输入(T1)。 上电复位输出(MRST)。 可编程逻辑阵列输出单元3 (PLAO[3])。
24	TCK	JTAG测试端口输入, 测试时钟。调试和下载访问。
25	TDO	JTAG测试端口输出, 测试数据输出。
26	P3.0/PWM0/PLAI[8]	通用输入和输出端口3.0 (P3.0)。 PWM相位0 (PWM0)。 可编程逻辑阵列输入单元8 (PLAI[8])。
27	P3.1/PWM1/PLAI[9]	通用输入和输出端口3.1 (P3.1)。 PWM相位1 (PWM1)。 可编程逻辑阵列输入单元9 (PLAI[9])。
28	P3.2/PWM2/PLAI[10]	通用输入和输出端口3.2 (P3.2)。 PWM相位2 (PWM2)。 可编程逻辑阵列输入单元10 (PLAI[10])。
29	P3.3/PWM3/PLAI[11]	通用输入和输出端口3.3 (P3.3)。 PWM相位3 (PWM3)。 可编程逻辑阵列输入单元11 (PLAI[11])。
30	P0.3/TRST/ADC _{BUSY}	通用输入和输出端口0.3 (P0.3)。 JTAG测试端口输入, 测试复位(TRST)。 JTAG reset input. 调试和下载访问。如果此引脚保持为低电平, 则无法访问JTAG, 因为JTAG接口处于复位状态并且P0.1/P0.2/P0.3均配置为GPIO引脚。
31	P3.4/PWM4/PLAI[12]	ADC _{BUSY} 信号输出(ADC _{BUSY})。 通用输入和输出端口3.4 (P3.4)。 PWM相位4 (PWM4)。 可编程逻辑阵列输入12 (PLAI[12])。
32	P3.5/PWM5/PLAI[13]	通用输入和输出端口3.5 (P3.5)。 PWM相位5 (PWM5)。 可编程逻辑阵列输入单元13 (PLAI[13])。
33	$\overline{\text{RST}}$	复位输入, 低电平有效。
34	IRQ0/P0.4/PWM _{TRIP} /PLAO[1]	多功能输入输出引脚。 外部中断请求0, 高电平有效(IRQ0)。 通用输入和输出端口0.4 (P0.4)。 PWM触发外部输入(PWM _{TRIP})。 可编程逻辑阵列输出单元1 (PLAO[1])。

ADuC7124/ADuC7126

引脚编号	引脚名称	描述
35	IRQ1/P0.5/ADC _{BUSY} /PLAO[2]	多功能输入输出引脚。 外部中断请求1, 高电平有效(IRQ1)。 通用输入和输出端口0.5 (P0.5)。 ADC _{BUSY} 信号输出(ADC _{BUSY})。 可编程逻辑阵列输出单元2 (PLAO[2])。
36	P2.0/SPM9/PLAO[5]/ $\overline{\text{CONV}}_{\text{START}}$ /SOUT0	通用输入和输出端口2.0 (P2.0)。 串行复用端口(SPM9)。 可编程逻辑阵列输出单元5 (PLAO[5])。 ADC开始转换输入信号($\overline{\text{CONV}}_{\text{START}}$)。 UART0输出(SOUT0)。
37	P0.7/ECLK/XCLK/SPM8/PLAO[4]/SINO	通用输入和输出端口0.7 (P0.7)。 外部时钟信号输出(ECLK)。 内部时钟发生器电路输入(XCLK)。 串行复用端口(SPM8)。 可编程逻辑阵列输出单元4 (PLAO[4])。 UART0输入(SINO)。
38	IOGND	GPIO地。通常连接到DGND。
39	IOV _{DD}	3.3 V电源, 用于GPIO和片内稳压器输入。
40	P3.6/PWM _{TRIP} /PLAI[14]	通用输入和输出端口3.6 (P3.6)。 PWM安全切断(PWM _{TRIP})。 可编程逻辑阵列输入单元14 (PLAI[14])。
41	P3.7/PWM _{SYNC} /PLAI[15]	通用输入和输出端口3.7 (P3.7)。 PWM同步输入/输出(PWM _{SYNC})。 可编程逻辑阵列输入单元15 (PLAI[15])。
42	P1.7/SPM7/DTR/ $\overline{\text{SPIC}}_{\text{S}}/PLAO[0]$	通用输入和输出端口1.7 (P1.7)。 串行复用端口。UART、SPI (SPM7)。 数据终端就绪(DTR)。 片选(SPICS)。 可编程逻辑阵列输出单元0 (PLAO[0])。
43	P1.6/SPM6/PLAI[6]	通用输入和输出端口1.6 (P1.6)。 串行复用端口(SPM6)。 可编程逻辑阵列输入单元6 (PLAI[6])。
44	P4.0/PLAO[8]/SIN1	通用输入和输出端口4.0 (P4.0)。 可编程逻辑阵列输出单元8 (PLAO[8])。 UART1输入(SIN1)。
45	P4.1/PLAO[9]/SOUT1	通用输入和输出端口4.1 (P4.1)。 可编程逻辑阵列输出单元9 (PLAO[9])。 UART1输出(SOUT1)。
46	P1.5/SPM5/DCD/SPIMISO/PLAI[5]/IRQ3	通用输入和输出端口1.5 (P1.5)。 串行复用端口。UART、SPI (SPM5)。 数据载波检测(DCD)。 主机输入、从机输出(SPI MISO)。 可编程逻辑阵列输入单元5 (PLAI[5])。
47	P1.4/SPM4/RI/SPICLK/PLAI[4]/IRQ2	外部中断请求3, 高电平有效(IRQ3)。 通用输入和输出端口1.4 (P1.4)。 串行复用端口。UART、SPI (SPM4)。 响铃指示(RI)。 串行时钟输入/输出(SPI SCLK)。 可编程逻辑阵列输入单元4 (PLAI[4])。
48	P1.3/SPM3/CTS/I2C1SDA/PLAI[3]	外部中断请求2, 高电平有效(IRQ2)。 通用输入和输出端口1.3 (P1.3)。 串行复用端口。UART、I2C1 (SPM3)。 清除发送(CTS)。 I2C1 (I2C1SDA)。 可编程逻辑阵列输入单元3 (PLAI[3])。
49	P1.2/SPM2/RTS/I2C1SCL/PLAI[2]	通用输入和输出端口1.2 (P1.2)。 串行复用端口(SPM2)。 准备发送(RTS)。 I2C1 (I2C1SCL)。 可编程逻辑阵列输入单元2 (PLAI[2])。

引脚编号	引脚名称	描述
50	P1.1/SPM1/SOUT0/I2C0SDA/PLAI[1]	通用输入和输出端口1.1 (P1.1)。串行复用端口(SPM1)。UART下载引脚、UART0输出(SOUT0)。I2C0 (I2C0SDA)。
51	P1.0/T1/SPM0/SIN0/I2C0SCL/PLAI[0]	可编程逻辑阵列输入单元1 (PLAI[1])。通用输入和输出端口1.0 (P1.0)。定时器1输入(T1)。串行复用端口(SPM0)。UART下载引脚、UART0输入(SIN0)。I2C0 (I2C0SCL)。
52	P4.2/PLAO[10]	可编程逻辑阵列输入单元0 (PLAI[0])。通用输入和输出端口4.2 (P4.2)。可编程逻辑阵列输出单元10 (PLAO[10])。
53	P4.3/PLAO[11]	通用输入和输出端口4.3 (P4.3)。可编程逻辑阵列输出单元11 (PLAO[11])。
54	P4.4/PLAO[12]	通用输入和输出端口4.4 (P4.4)。可编程逻辑阵列输出单元12 (PLAO[12])。
55	RTCK	JTAG测试端口输出, JTAG返回测试时钟。
56	V _{REF}	2.5 V内部基准电压。使用内部基准电压源时必须连接至一个0.47 μF电容。
57	DAC _{REF}	DAC外部基准电压。范围: DACGND至DACV _{DD} 。
58	AV _{DD}	3.3 V模拟电源。
59	AGND	模拟地。模拟电路的地基准点。
60	GND _{REF}	ADC地基准电压。为了优化性能, 模拟电源应同IOGND和DGND分离。
61	ADC0	单端或差分模拟输入0。
62	ADC1	单端或差分模拟输入1。
63	ADC2/CMP0	单端或差分模拟输入2 (ADC2)。比较器正输入(CMP0)。
64	ADC3/CMP1	单端或差分模拟输入3 (ADC3)。比较器负输入(CMP1)。

ADuC7124/ADuC7126

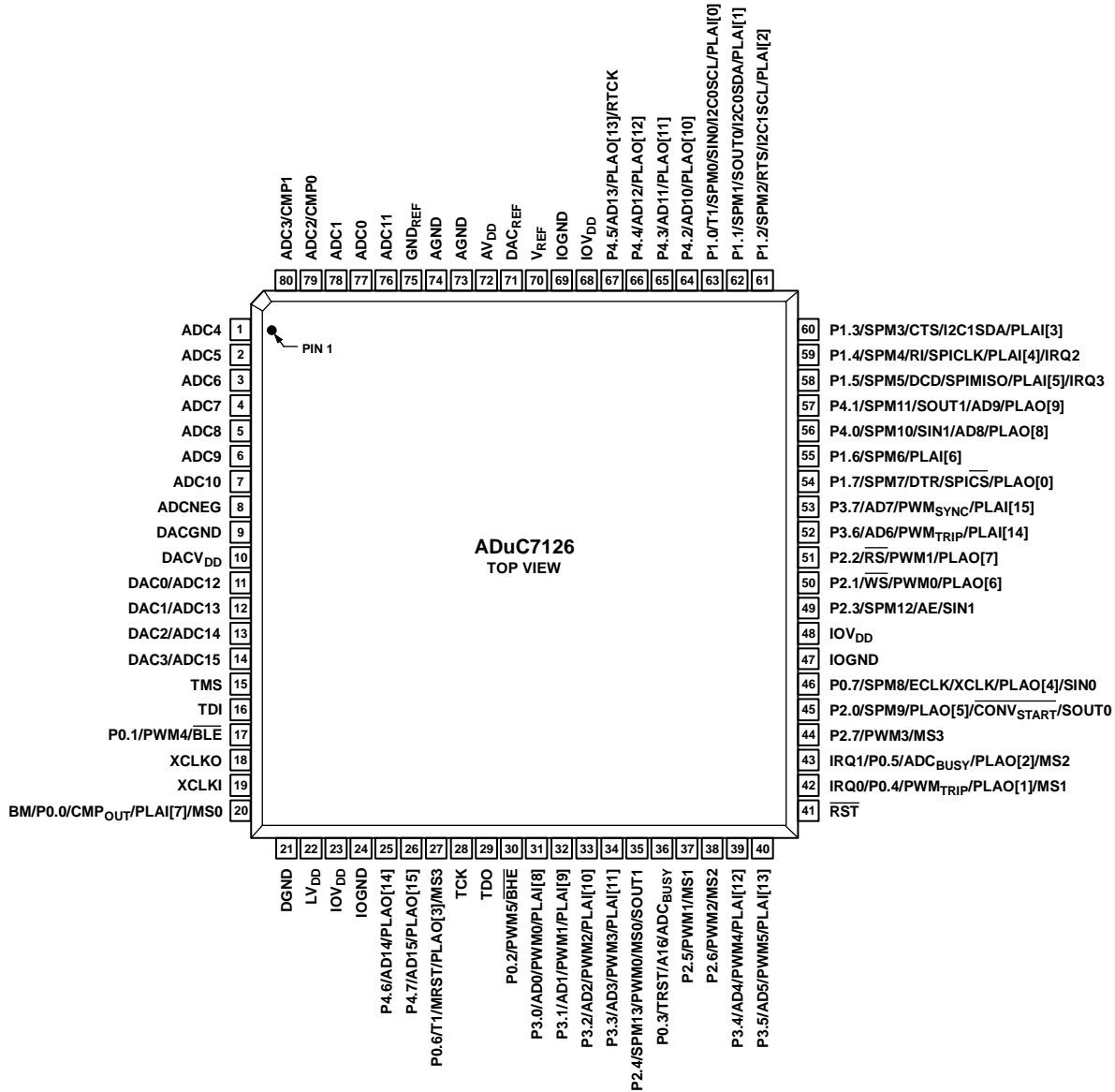


图8. ADuC7126引脚配置

表10. 引脚功能描述(ADuC7126 80引脚LQFP)

引脚编号	引脚名称	描述
1	ADC4	单端或差分模拟输入4。
2	ADC5	单端或差分模拟输入5。
3	ADC6	单端或差分模拟输入6。
4	ADC7	单端或差分模拟输入7。
5	ADC8	单端或差分模拟输入8。
6	ADC9	单端或差分模拟输入9。
7	ADC10	单端或差分模拟输入10。
8	ADCNEG	伪差分模式下ADC偏置点或反相模拟输入。必须连接到要转换信号的地。该偏置点必须在0V至1V之间。
9	DACGND	DAC地。通常连接到AGND。
10	DACV _{DD}	DAC 3V电压源。必须连接到AV _{DD} 。

引脚编号	引脚名称	描述
11	DAC0/ADC12	DAC0电压输出(DAC0)。单端或差分模拟输入12 (ADC12)。
12	DAC1/ADC13	DAC1电压输出(DAC1)。单端或差分模拟输入13 (ADC13)。
13	DAC2/ADC14	DAC2电压输出(DAC2)。单端或差分模拟输入14 (ADC14)。
14	DAC3/ADC15	DAC3电压输出(DAC3)。单端或差分模拟输入15 (ADC15)。
15	TMS	JTAG测试端口输入, 测试模式选择。调试和下载访问。
16	TDI	JTAG测试端口输入, 测试数据输入。调试和下载访问。
17	P0.1/PWM4/ $\overline{\text{BLE}}$	通用输入和输出端口0.1 (P0.1)。PWM相位4 (PWM4)。外部存储器字节低电平使能(BLE)。
18	XCLKO	晶体振荡反相器输出。
19	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。
20	BM/P0.0/CMP _{OUT} /PLAI[7]/MS0	多功能输入输出引脚。引导模式入口引脚(BM)。复位时, 如果BM为低电平, 则ADuC7126进入UART下载模式; 如果BM通过1 k Ω 电阻拉高, 则ADuC7126执行代码。如果复位时BM处于低电平, 且闪存地址0x800014 = 0xFFFFFFFF, 则I2C版本的ADuC7026进入I ² C下载模式。如果复位时BM被拉高, 或者如果复位时BM处于低电平, 但闪存地址0x800014不等于0xFFFFFFFF, 则ADuC7026执行代码。通用输入和输出端口0.0 (P0.0)。电压比较器输出/可编程逻辑阵列输入元件7 (CMP _{OUT})。外部存储器选择0 (MS0)。默认情况下, 此引脚配置为GPIO。
21	DGND	内核逻辑地。
22	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47 μ F电容器连接至DGND。
23	IOV _{DD}	3.3 V电源, 用于GPIO和片内稳压器输入。
24	I0GND	GPIO地。通常连接到DGND。
25	P4.6/AD14/PLAO[14]	通用输入和输出端口4.6 (P4.6)。外部存储器接口(AD14)。可编程逻辑阵列输出单元14 (PLAO[14])。
26	P4.7/AD15/PLAO[15]	通用输入和输出端口4.7 (P4.7)。外部存储器接口(AD15)。可编程逻辑阵列输出单元15 (PLAO[15])。
27	P0.6/T1/MRST/PLAO[3]/MS3	多功能引脚, 复位后输出低电平。通用输出端口0.6 (P0.6)。定时器1输入(T1)。上电复位输出(MRST)。可编程逻辑阵列输出单元3 (PLAO[3])。外部存储器选择3 (MS3)。
28	TCK	JTAG测试端口输入, 测试时钟。调试和下载访问。
29	TDO	JTAG测试端口输出, 测试数据输出。调试和下载访问。
30	P0.2/PWM5/ $\overline{\text{BHE}}$	通用输入和输出端口0.2 (P0.2)。PWM相位5 (PWM5)。外部存储器字节高电平使能($\overline{\text{BHE}}$)。
31	P3.0/AD0/PWM0/PLAI[8]	通用输入和输出端口3.0 (P3.0)。外部存储器接口(AD0)。PWM相位0 (PWM0)。可编程逻辑阵列输入单元8 (PLAI[8])。
32	P3.1/AD1/PWM1/PLAI[9]	通用输入和输出端口3.1 (P3.1)。外部存储器接口(AD1)。PWM相位1 (PWM1)。可编程逻辑阵列输入单元9 (PLAI[9])。
33	P3.2/AD2/PWM2/PLAI[10]	通用输入和输出端口3.2 (P3.2)。外部存储器接口(AD2)。PWM相位2 (PWM2)。可编程逻辑阵列输入单元10 (PLAI[10])。

ADuC7124/ADuC7126

引脚编号	引脚名称	描述
34	P3.3/AD3/PWM3/PLAI[11]	通用输入和输出端口3.3 (P3.3)。 外部存储器接口(AD3)。 PWM相位3 (PWM3)。 可编程逻辑阵列输入单元11 (PLAI[11])。
35	P2.4/SPM13/PWM0/MS0/SOUT1	通用输入和输出端口2.4 (P2.4)。 串行复用端口(SPM13)。 PWM相位0 (PWM0)。 外部存储器选择0 (MS0)。 UART1输出(SOUT1)。
36	P0.3/TRST/A16/ADC _{BUSY}	通用输入和输出端口0.3 (P0.3)。 JTAG测试端口输入, 测试复位(TRST)。JTAG复位输入。调试和下载访问。如果此引脚保持为低电平, 则无法访问JTAG, 因为JTAG接口处于复位状态并且P0.1/P0.2/P0.3均配置为GPIO引脚。 地址线(A16)。 ADC _{BUSY} 信号输出(ADC _{BUSY})。
37	P2.5/PWM1/MS1	通用输入和输出端口2.5 (P2.5)。 PWM相位1 (PWM1)。 外部存储器选择1 (MS1)。
38	P2.6/PWM2/MS2	通用输入和输出端口2.6 (P2.6)。 PWM相位2 (PWM2)。 外部存储器选择2 (MS2)。
39	P3.4/AD4/PWM4/PLAI[12]	通用输入和输出端口3.4 (P3.4)。 外部存储器接口(AD4)。 PWM相位4 (PWM4)。
40	P3.5/AD5/PWM5/PLAI[13]	可编程逻辑阵列输入12 (PLAI[12])。 通用输入和输出端口3.5 (P3.5)。 外部存储器接口(AD5)。 PWM相位5 (PWM5)。
41	$\overline{\text{RST}}$	可编程逻辑阵列输入单元13 (PLAI[13])。 复位输入, 低电平有效。
42	IRQ0/P0.4/PWM _{TRIP} /PLAO[1]/MS1	多功能输入输出引脚。 外部中断请求0, 高电平有效(IRQ0)。 通用输入和输出端口0.4 (P0.4)。 PWM触发外部输入(PWM _{TRIP})。 可编程逻辑阵列输出单元1 (PLAO[1])。 外部存储器选择1 (MS1)。
43	IRQ1/P0.5/ADC _{BUSY} /PLAO[2]/MS2	多功能输入输出引脚。 外部中断请求1, 高电平有效(IRQ1)。 通用输入和输出端口0.5 (P0.5)。 ADC _{BUSY} 信号输出(ADC _{BUSY})。 可编程逻辑阵列输出单元2 (PLAO[2])。 外部存储器选择2 (MS2)。
44	P2.7/PWM3/MS3	通用输入和输出端口2.7 (P2.7)。 PWM相位3 (PWM3)。 外部存储器选择3 (MS3)。
45	P2.0/SPM9/PLAO[5]/CONV _{START} /SOUT0	通用输入和输出端口2.0 (P2.0)。 串行复用端口(SPM9)。 可编程逻辑阵列输出单元5 (PLAO[5])。 ADC开始转换输入信号(CONV _{START})。 UART0输出(SOUT0)。
46	P0.7/SPM8/ECLK/XCLK/PLAO[4]/SINO	通用输入和输出端口0.7 (P0.7)。 串行复用端口(SPM8)。 外部时钟信号输出(ECLK)。 内部时钟发生器电路输入(XCLK)。 可编程逻辑阵列输出单元4 (PLAO[4])。 UART0输入(SINO)。
47	IOGND	GPIO地。通常连接到DGND。
48	IOV _{DD}	3.3 V电源, 用于GPIO和片内稳压器输入。

引脚编号	引脚名称	描述
49	P2.3/SPM12/AE/SIN1	通用输入和输出端口2.3 (P2.3)。 串行复用端口(SPM12)。 外部存储器访问使能(AE)。 UART1输入(SIN1)。
50	P2.1/ \overline{WS} /PWM0/PLAO[6]	通用输入和输出端口2.1 (P2.1)。 外部存储器写入选通(\overline{WS})。 PWM相位0 (PWM0)。 可编程逻辑阵列输出单元6 (PLAO[6])。
51	P2.2/ \overline{RS} /PWM1/PLAO[7]	通用输入和输出端口2.2 (P2.2)。 外部存储器写入选通(\overline{RS})。 PWM相位1 (PWM1)。 可编程逻辑阵列输出单元7 (PLAO[7])。
52	P3.6/AD6/PWM _{TRIP} /PLAI[14]	通用输入和输出端口3.6 (P3.6)。 外部存储器接口(AD6)。 PWM安全切断(PWM _{TRIP})。 可编程逻辑阵列输入单元14 (PLAI[14])。
53	P3.7/AD7/PWM _{SYNC} /PLAI[15]	通用输入和输出端口3.7 (P3.7)。 外部存储器接口(AD7)。 PWM同步(PWM _{SYNC})。 可编程逻辑阵列输入单元15 (PLAI[15])。
54	P1.7/SPM7/DTR/ \overline{SPICS} /PLAO[0]	通用输入和输出端口1.7 (P1.7)。 串行复用端口(SPM7)。 数据终端就绪(DTR)。 片选(SPICS)。 可编程逻辑阵列输出单元0 (PLAO[0])。
55	P1.6/SPM6/PLAI[6]	通用输入和输出端口1.6 (P1.6)。 串行复用端口(SPM6)。 可编程逻辑阵列输入单元6 (PLAI[6])。
56	P4.0/SPM10/SIN1/AD8/PLAO[8]	通用输入和输出端口4.0 (P4.0)。 串行复用端口(SPM10)。 UART1输入(SIN1)。 外部存储器接口(AD8)。 可编程逻辑阵列输出单元8 (PLAO[8])。
57	P4.1/SPM11/SOUT1/AD9/PLAO[9]	通用输入和输出端口4.1 (P4.1)。 串行复用端口(SPM11)。 UART1输出(SOUT1)。 外部存储器接口(AD9)。 可编程逻辑阵列输出单元9 (PLAO[9])。
58	P1.5/SPM5/DCD/SPIMISO/PLAI[5]/IRQ3	通用输入和输出端口1.5 (P1.5)。 串行复用端口(SPM5)。 数据载波检测(DCD)。 主机输入、从机输出(SPI MISO)。 可编程逻辑阵列输入单元5 (PLAI[5])。
59	P1.4/SPM4/RI/SPICLK/PLAI[4]/IRQ2	外部中断请求3, 高电平有效(IRQ3)。 通用输入和输出端口1.4 (P1.4)。 串行复用端口(SPM4)。 响铃指示(RI)。 串行时钟输入/输出(SPI SCLK)。 可编程逻辑阵列输入单元4 (PLAI[4])。
60	P1.3/SPM3/CTS/I2C1SDA/PLAI[3]	外部中断请求2, 高电平有效(IRQ2)。 通用输入和输出端口1.3 (P1.3)。 串行复用端口(SPM3)。 清除发送(CTS)。 I2C1 (I2C1SDA)。 可编程逻辑阵列输入单元3 (PLAI[3])。
61	P1.2/SPM2/RTS/I2C1SCL/PLAI[2]	通用输入和输出端口1.2 (P1.2)。 串行复用端口(SPM2)。 准备发送(RTS)。 I2C1 (I2C1SCL)。 可编程逻辑阵列输入单元2 (PLAI[2])。

ADuC7124/ADuC7126

引脚编号	引脚名称	描述
62	P1.1/SPM1/SOUT0/I2C0SDA/PLAI[1]	通用输入和输出端口1.1 (P1.1)。 串行复用端口(SPM1)。 UART0输出(SOUT0)。 I2C0 (I2C0SDA)。 可编程逻辑阵列输入单元1 (PLAI[1])。
63	P1.0/T1/SPM0/SIN0/I2C0SCL/PLAI[0]	通用输入和输出端口1.0 (P1.0)。 定时器1输入(T1)。 串行复用端口(SPM0)。 UART0输入(SIN0)。 I2C0 (I2C0SCL)。 可编程逻辑阵列输入单元0 (PLAI[0])。
64	P4.2/AD10/PLAO[10]	通用输入和输出端口4.2 (P4.2)。 外部存储器接口(AD10)。 可编程逻辑阵列输出单元10 (PLAO[10])。
65	P4.3/AD11/PLAO[11]	通用输入和输出端口4.3 (P4.3)。 外部存储器接口(AD11)。 可编程逻辑阵列输出单元11 (PLAO[11])。
66	P4.4/AD12/PLAO[12]	通用输入和输出端口4.4 (P4.4)。 外部存储器接口(AD12)。 可编程逻辑阵列输出单元12 (PLAO[12])。
67	P4.5/AD13/PLAO[13]/RTCK	通用输入和输出端口4.5 (P4.5)。 外部存储器接口(AD13)。 可编程逻辑阵列输出单元13 (PLAO[13])。 JTAG返回测试时钟(RTCK)。
68	IOV _{DD}	3.3 V电源, 用于GPIO和片内稳压器输入。
69	IOGND	GPIO地。通常连接到DGND。
70	V _{REF}	2.5 V内部基准电压。使用内部基准电压源时必须连接至一个0.47 μF电容。
71	DAC _{REF}	DAC外部基准电压。范围: DACGND至DACV _{DD} 。
72	AV _{DD}	3.3 V模拟电源。
73, 74	AGND	模拟地。模拟电路的地基准点。
75	GND _{REF}	ADC地基准电压。为了优化性能, 模拟电源应同IOGND和DGND分离。
76	ADC11	单端或差分模拟输入11。
77	ADC0	单端或差分模拟输入0。
78	ADC1	单端或差分模拟输入1。
79	ADC2/CMP0	单端或差分模拟输入2 (ADC2)。 比较器正输入(CMP0)。
80	ADC3/CMP1	单端或差分模拟输入3 (ADC3)。 比较器负输入(CMP1)。

典型工作特性

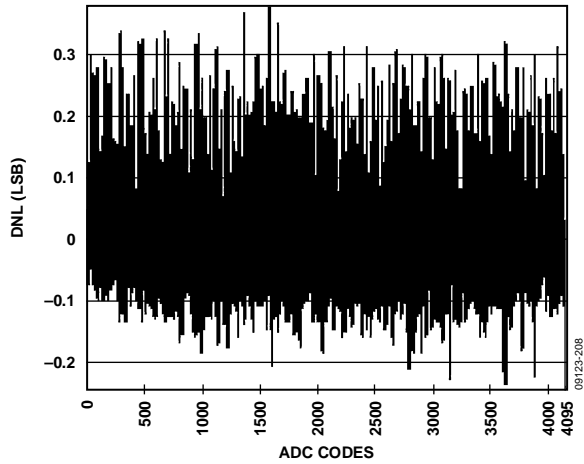


图9. 典型DNL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = ADC0$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.38 LSB, 代码1567
 最差情况负值 = -0.24 LSB, 代码4094

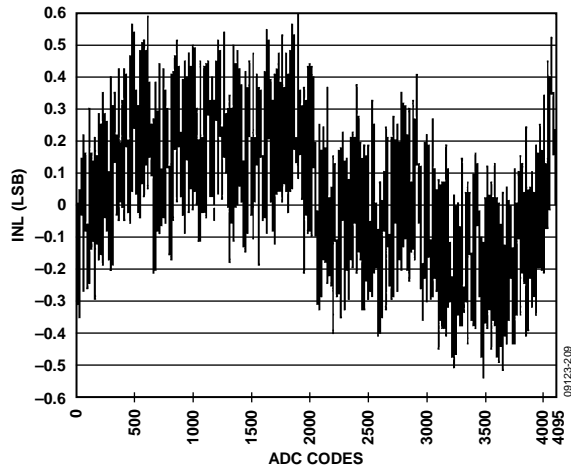


图10. 典型INL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = ADC0$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.60 LSB, 代码1890
 最差情况负值 = -0.54 LSB, 代码3485

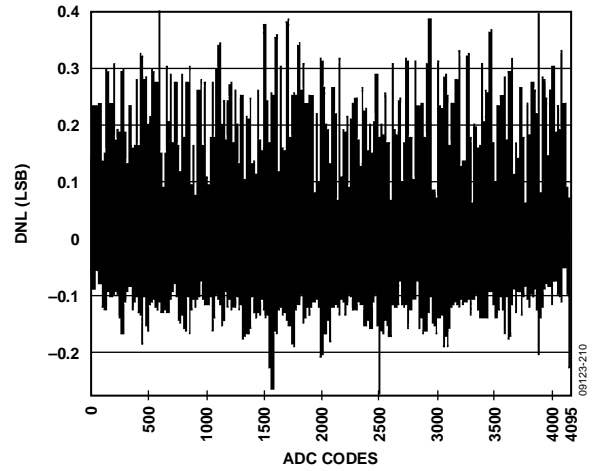


图11. 典型DNL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = DAC1/ADC13$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.40 LSB, 代码607
 最差情况负值 = -0.27 LSB, 代码2486

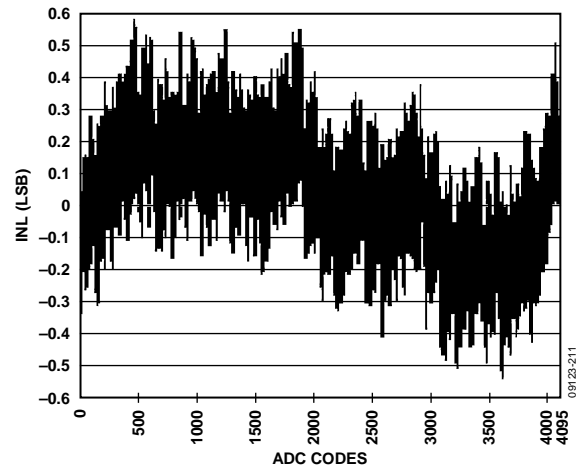


图12. 典型INL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = DAC1/ADC13$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.58 LSB, 代码480
 最差情况负值 = -0.54 LSB, 代码3614

ADuC7124/ADuC7126

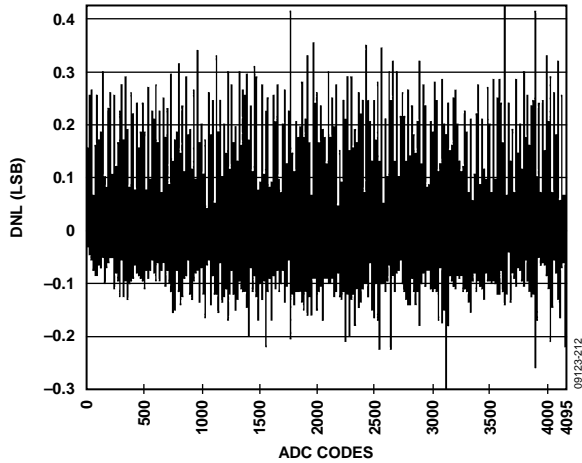


图13. 典型DNL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = ADC8$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.42 LSB, 代码3583
 最差情况负值 = -0.32 LSB, 代码3073

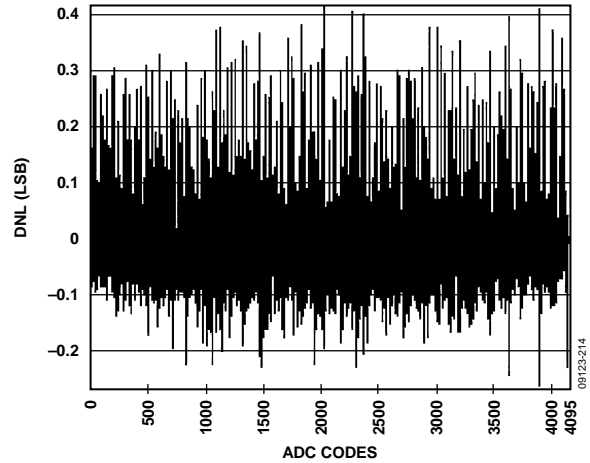


图15. 典型DNL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = DAC3/ADC15$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.41 LSB, 代码2016
 最差情况负值 = -0.26 LSB, 代码3841

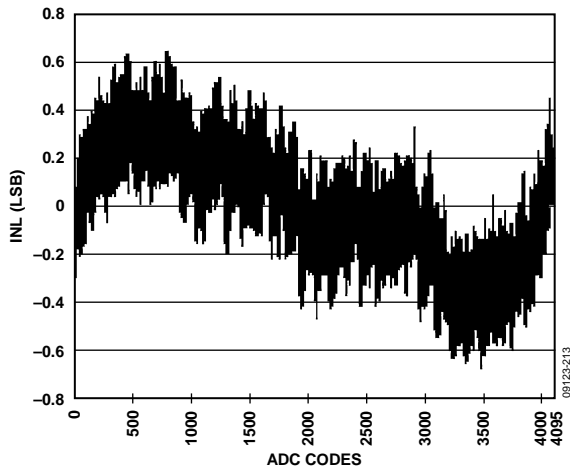


图14. 典型INL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = ADC8$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.64 LSB, 代码802
 最差情况负值 = -0.69 LSB, 代码3485

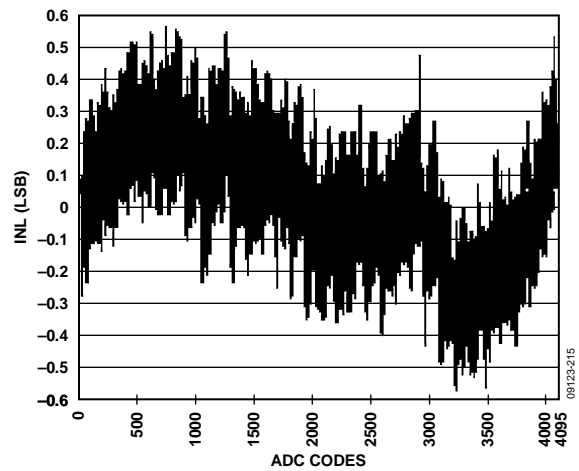


图16. 典型INL误差,
 温度25°C, V_{REF} = 内部2.5 V, 单端模式
 $ADCCP = DAC3/ADC15$, $ADCCN = ADC0$, 采样速率 = 345 kHz
 最差情况正值 = 0.55 LSB, 代码738
 最差情况负值 = -0.68 LSB, 代码3230

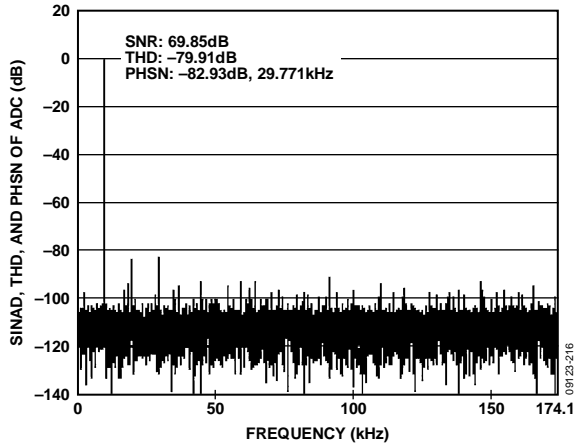


图17. ADC的SINAD、THD和PHSN,
 V_{REF} = 内部2.5 V, 单端模式
 ADCCP = ADC0

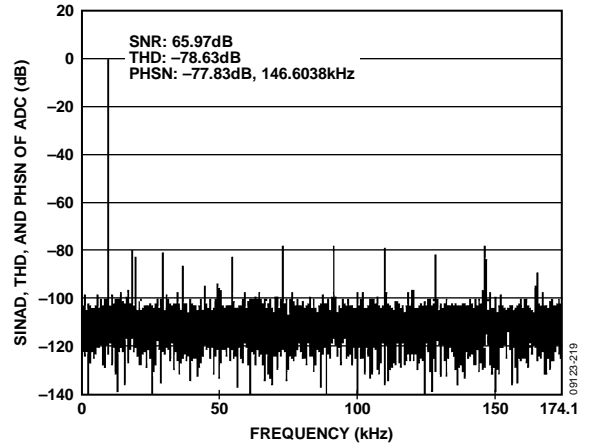


图20. ADC的SINAD、THD和PHSN,
 V_{REF} = 内部2.5 V, 单端模式
 ADCCP = ADC15/DAC3, ADCCN = ADC0

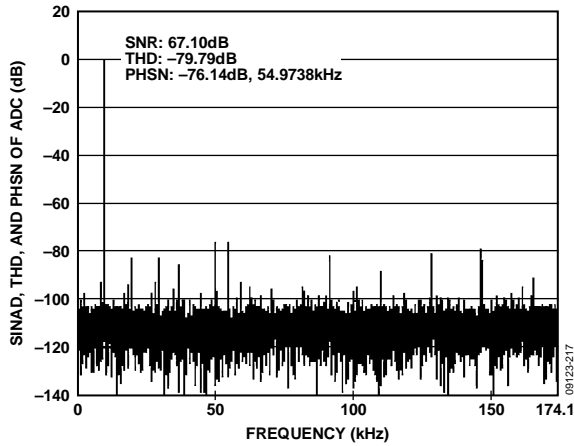


图18. ADC的SINAD、THD和PHSN,
 V_{REF} = 内部2.5 V, 单端模式
 ADCCP = DAC1/ADC13, ADCCN = ADC0

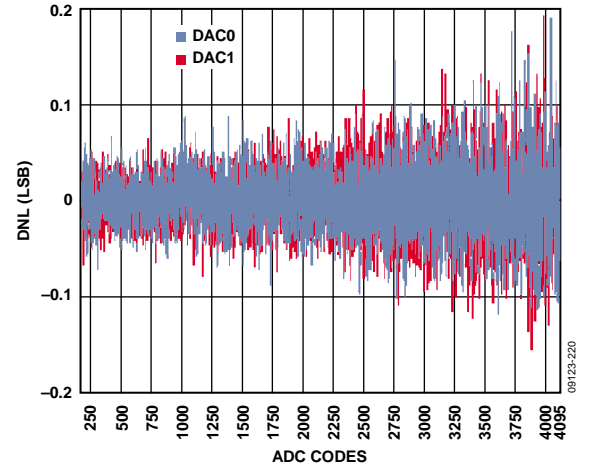


图21. DAC DNL误差
 DAC0最大正DNL: 0.188951, DAC1最大正DNL: 0.190343
 DAC0最大负DNL: -0.120081, DAC1最大负DNL: -0.15697

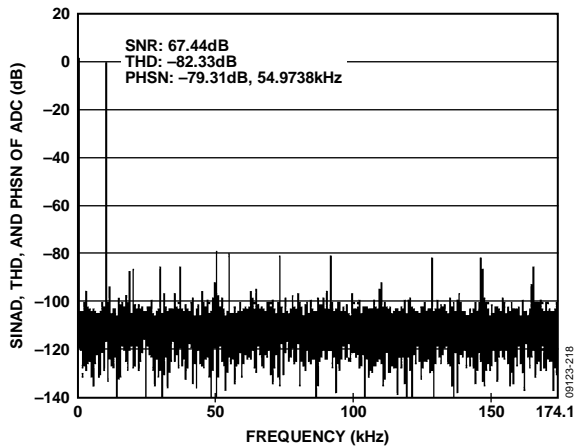


图19. ADC的SINAD、THD和PHSN,
 V_{REF} = 内部2.5 V, 单端模式
 ADCCP = ADC8, ADCCN = ADC0

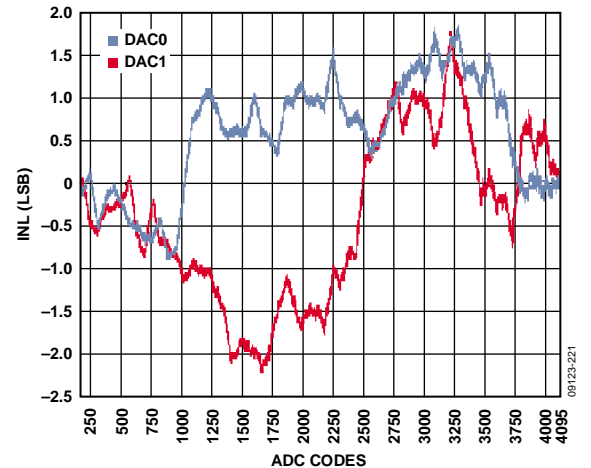


图22. DAC INL误差
 DAC0最大正INL: 1.84106, DAC1最大正INL: 1.75312
 DAC0最大负INL: -0.887319, DAC1最大负INL: -2.23708

术语

ADC技术规格

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...000到0000...001)的跃变点与理想点 $+\frac{1}{2}$ LSB之间的偏差。

增益误差

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。

信号与(噪声+失真)比

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，信号与噪声+失真比值的理论计算值为：

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

因此，对于12位转换器，该值为74dB。

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采用ARM7内核，具有4种附加特性。

- T: 支持Thumb[®] (16-bit)指令集。
- D: 支持调试。
- M: 支持长乘。
- I: 包含一个支持嵌入式系统调试的EmbeddedICE模块。

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，处理器中断，并进入调试状态。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。

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23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。

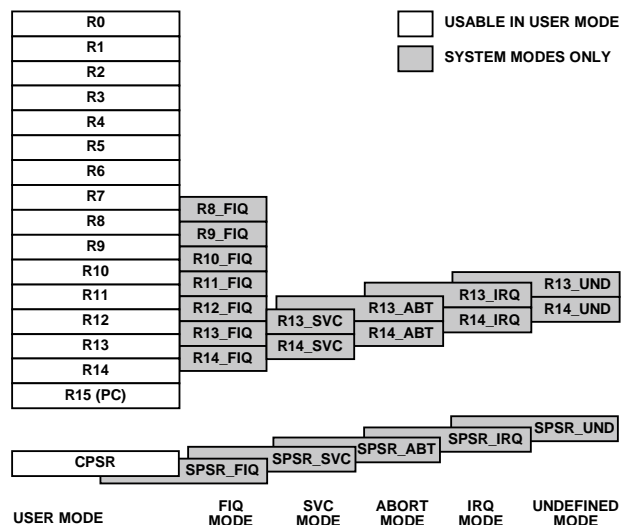


图23. 寄存器结构图

ADuC7124/ADuC7126

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：

- DDI0029G, ARM7TDMI技术参考手册
- DDI-0100, ARM架构参考手册

中断延迟

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41.78MHz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

ADuC7124/ADuC7126有三个独立存储器模块：一个32 kB SRAM和两个64 kB片内Flash/EE存储器模块。片内Flash/EE存储器有126 kB可供用户使用，剩余2 kB保留供系统内核使用。这些存储器模块的映射如图24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。通过REMAPMMR的Bit0位置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。

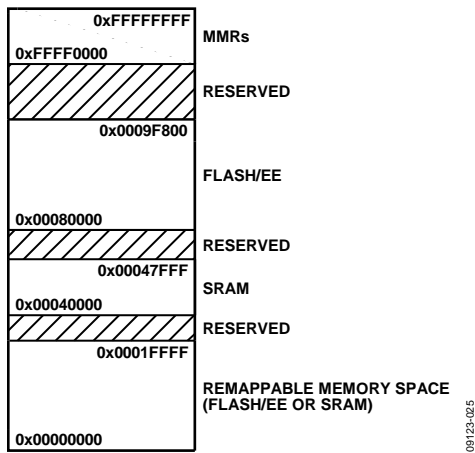


图24. 物理存储器映射图

存储器访问

ARM7内核把存储器看成是 2^{32} 个字节的一个线性阵列。不同的存储器模块映射如图24所示。

ADuC7124/ADuC7126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。

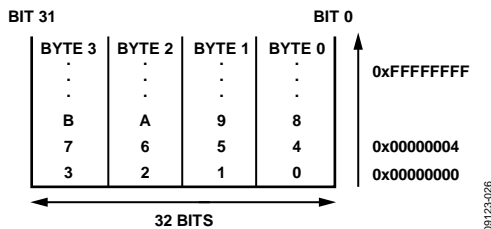


图25. 从小到大顺序格式

FLASH/EE存储器

128 kB的Flash/EE分为两个32 kB × 16 bit的模块。第一个模块中，31 kB × 16 bit是用户空间，1 kB × 16 bit保留用于工厂配置的引导页面。Flash/EE的页面大小为512个字节。

第二个64 kB模块的组织方式相似，配置为32 kB × 16 bit。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模式下每执行一个指令(32位指令)，必须读取两次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41.78MHz，而相应的以全ARM模式为20.89MHz(参见“SRAM和Flash/EE执行时间”部分)。

SRAM

用户可以使用32 kB的SRAM，它的组织形式为8 kB × 32 bit，即16 kB字。如果SRAM被配置成32位宽的存储器阵列，ARM代码可以直接在SRAM中以41.78MHz的速度执行(参见“SRAM和Flash/EE执行时间”部分)。

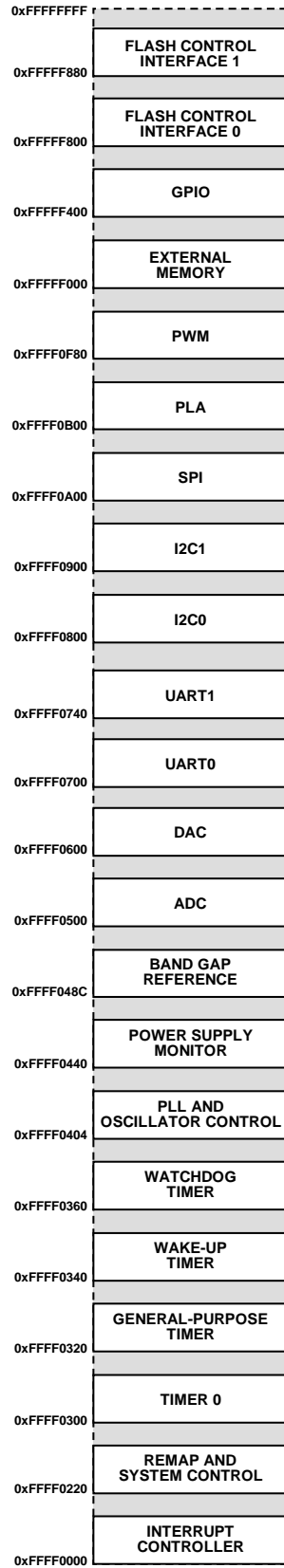
存储器映射寄存器

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

MMR空间为CPU和所有片内外设提供接口。除了内核寄存器，所有的寄存器都位于寄存器区域内。图26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11至表29为所有寄存器存储器映射。

读取或写入一个寄存器所需的存取时间取决于高级微控制器总线结构(AMBA)总线，该总线可用来访问外围设备。处理器有两个AMBA总线：高性能总线(AHB)用于系统模块，高级外围总线(APB)用于低性能外围设备。访问AHB需要一个周期，访问APB需要两个周期。除了Flash/EE存储器和通用输入输出端口以外，ADuC7124/ADuC7126中的所有外围设备均位于APB上。

ADuC7124/ADuC7126



09123-010

图 26. 存储器映射寄存器

表11. IRQ基地址 = 0xFFFF0000

地址	名称	字节	访问类型
0xFFFF0000	IRQSTA	4	R
0xFFFF0004	IRQSIG	4	R
0xFFFF0008	IRQEN	4	R/W
0xFFFF000C	IRQCLR	4	W
0xFFFF0010	SWICFG	4	W
0xFFFF0014	IRQBASE	4	R/W
0xFFFF001C	IRQVEC	4	R
0xFFFF0020	IRQP0	4	R/W
0xFFFF0024	IRQP1	4	R/W
0xFFFF0028	IRQP2	4	R/W
0xFFFF002C	IRQP3	4	R/W
0xFFFF0030	IRQCONN	1	R/W
0xFFFF0034	IRQCONE	4	R/W
0xFFFF0038	IRQCLRE	1	W
0xFFFF003C	IRQSTAN	1	R/W
0xFFFF0100	FIQSTA	4	R
0xFFFF0104	FIQSIG	4	R
0xFFFF0108	FIQEN	4	R/W
0xFFFF010C	FIQCLR	4	W
0xFFFF011C	FIQVEC	4	R
0xFFFF013C	FIQSTAN	1	R/W

表12. 系统控制基地址 = 0xFFFF0200

地址	名称	字节	访问类型
0xFFFF0220	REMAP	1	R/W
0xFFFF0230	RSTSTA	1	R
0xFFFF0234	RSTCLR	1	W
0xFFFF0248	RSTKEY0	1	W
0xFFFF024C	RSTCFG	1	R/W
0xFFFF0250	RSTKEY1	1	W

表13. 定时器基地址 = 0xFFFF0300

地址	名称	字节	访问类型
0xFFFF0300	T0LD	2	R/W
0xFFFF0304	T0VAL	2	R
0xFFFF0308	T0CON	2	R/W
0xFFFF030C	T0CLRI	1	W
0xFFFF0320	T1LD	4	R/W
0xFFFF0324	T1VAL	4	R
0xFFFF0328	T1CON	2	R/W
0xFFFF032C	T1CLRI	1	W
0xFFFF0330	T1CAP	4	R/W
0xFFFF0340	T2LD	4	R/W
0xFFFF0344	T2VAL	4	R
0xFFFF0348	T2CON	2	R/W
0xFFFF034C	T2CLRI	1	W
0xFFFF0360	T3LD	2	R/W
0xFFFF0364	T3VAL	2	R
0xFFFF0368	T3CON	2	R/W
0xFFFF036C	T3CLRI	1	W

ADuC7124/ADuC7126

表14. PLL/PSM基地址 = 0xFFFF0400

地址	名称	字节	访问类型
0xFFFF0404	POWKEY1	2	W
0xFFFF0408	POWCON0	1	R/W
0xFFFF040C	POWKEY2	2	W
0xFFFF0410	PLLKEY1	4	W
0xFFFF0414	PLLCON	1	R/W
0xFFFF0418	PLLKEY2	4	W
0xFFFF0434	POWKEY3	2	W
0xFFFF0438	POWCON1	2	R/W
0xFFFF043C	POWKEY4	2	W

表15. PSM基地址 = 0xFFFF0440

地址	名称	字节	访问类型
0xFFFF0440	PSMCON	2	R/W
0xFFFF0444	CMPCON	2	R/W

表16. 基准电压基地址 = 0xFFFF0480

地址	名称	字节	访问类型
0xFFFF048C	REFCON	1	R/W

表17. ADC基地址 = 0xFFFF0500

地址	名称	字节	访问类型
0xFFFF0500	ADCCON	2	R/W
0xFFFF0504	ADCCP	1	R/W
0xFFFF0508	ADCCN	1	R/W
0xFFFF050C	ADCSTA	1	R
0xFFFF0510	ADCDAT	4	R
0xFFFF0514	ADCRST	1	R/W
0xFFFF0530	ADCGN	2	R/W
0xFFFF0534	ADCOF	2	R/W
0xFFFF0544	TSCON	1	R/W
0xFFFF0548	TEMPREF	2	R/W

表18. DAC基地址 = 0xFFFF0600

地址	名称	字节	访问类型
0xFFFF0600	DAC0CON	1	R/W
0xFFFF0604	DAC0DAT	4	R/W
0xFFFF0608	DAC1CON	1	R/W
0xFFFF060C	DAC1DAT	4	R/W
0xFFFF0610	DAC2CON	1	R/W
0xFFFF0614	DAC2DAT	4	R/W
0xFFFF0618	DAC3CON	1	R/W
0xFFFF061C	DAC3DAT	4	R/W
0xFFFF0650	DACBKEY1	2	W
0xFFFF0654	DACBCFG	1	R/W
0xFFFF0658	DACBKEY2	2	W

表19. UART0基地址 = 0xFFFF0700

地址	名称	字节	访问类型	周期
0xFFFF0700	COM0TX	1	R/W	2
0xFFFF0700	COM0RX	1	R	2
0xFFFF0700	COM0DIV0	1	R/W	2
0xFFFF0704	COM0IEN0	1	R/W	2
0xFFFF0704	COM0DIV1	1	R/W	2
0xFFFF0708	COM0IID0	1	R	2
0xFFFF0708	COM0FCR	1	R/W	2
0xFFFF070C	COM0CON0	1	R/W	2
0xFFFF0710	COM0CON1	1	R/W	2
0xFFFF0714	COM0STA0	2	R	2
0xFFFF0718	COM0STA1	2	R	2
0xFFFF072C	COM0DIV2	2	R/W	2

表20. UART1基地址 = 0xFFFF0740

地址	名称	字节	访问类型	周期
0xFFFF0740	COM1TX	1	R/W	2
0xFFFF0740	COM1RX	1	R	2
0xFFFF0740	COM1DIV0	1	R/W	2
0xFFFF0744	COM1IEN0	1	R/W	2
0xFFFF0744	COM1DIV1	1	R/W	2
0xFFFF0748	COM1IID0	1	R	2
0xFFFF0748	COM1FCR	1	R/W	2
0xFFFF074C	COM1CON0	1	R/W	2
0xFFFF0750	COM1CON1	1	R/W	2
0xFFFF0754	COM1STA0	2	R	2
0xFFFF0758	COM1STA1	2	R	2
0xFFFF076C	COM1DIV2	2	R/W	2

表21. I2C0基地址 = 0xFFFF0800

地址	名称	字节	访问类型	周期
0xFFFF0800	I2C0MCON	2	R/W	2
0xFFFF0804	I2C0MSTA	2	R	2
0xFFFF0808	I2C0MRX	1	R	2
0xFFFF080C	I2C0MTX	2	R/W	2
0xFFFF0810	I2C0MCNT0	2	R/W	2
0xFFFF0814	I2C0MCNT1	1	R	2
0xFFFF0818	I2C0ADRO	1	R/W	2
0xFFFF081C	I2C0ADR1	1	R/W	2
0xFFFF0824	I2C0DIV	2	R/W	2
0xFFFF0828	I2C0SCON	2	R/W	2
0xFFFF082C	I2C0SSTA	2	R	2
0xFFFF0830	I2C0SRX	1	R	2
0xFFFF0834	I2C0STX	1	W	2
0xFFFF0838	I2C0ALT	1	R/W	2
0xFFFF083C	I2C0ID0	1	R/W	2
0xFFFF0840	I2C0ID1	1	R/W	2
0xFFFF0844	I2C0ID2	1	R/W	2
0xFFFF0848	I2C0ID3	1	R/W	2
0xFFFF084C	I2C0FSTA	1	R/W	2

ADuC7124/ADuC7126

表22. I2C1基础地址 = 0xFFFF0900

地址	名称	字节	访问类型	周期
0xFFFF0900	I2C1MCON	2	R/W	2
0xFFFF0904	I2C1MSTA	2	R	2
0xFFFF0908	I2C1MRX	1	R	2
0xFFFF090C	I2C1MTX	2	R/W	2
0xFFFF0910	I2C1MCNT0	2	R/W	2
0xFFFF0914	I2C1MCNT1	1	R	2
0xFFFF0918	I2C1ADRO	1	R/W	2
0xFFFF091C	I2C1ADR1	1	R/W	2
0xFFFF0924	I2C1DIV	2	R/W	2
0xFFFF0928	I2C1SCON	2	R/W	2
0xFFFF092C	I2C1SSTA	2	R	2
0xFFFF0930	I2C1SRX	1	R	2
0xFFFF0934	I2C1STX	1	W	2
0xFFFF0938	I2C1ALT	1	R/W	2
0xFFFF093C	I2C1ID0	1	R/W	2
0xFFFF0940	I2C1ID1	1	R/W	2
0xFFFF0944	I2C1ID2	1	R/W	2
0xFFFF0948	I2C1ID3	1	R/W	2
0xFFFF094C	I2C1FSTA	1	R/W	2

表23. SPI基础地址 = 0xFFFF0A00

地址	名称	字节	访问类型	周期
0xFFFF0A00	SPISTA	2	R	2
0xFFFF0A04	SPIRX	1	R	2
0xFFFF0A08	SPITX	1	W	2
0xFFFF0A0C	SPIDIV	1	R/W	2
0xFFFF0A10	SPICON	2	R/W	2

表24. PLA基础地址 = 0xFFFF0B00

地址	名称	字节	访问类型	周期
0xFFFF0B00	PLAELM0	2	R/W	2
0xFFFF0B04	PLAELM1	2	R/W	2
0xFFFF0B08	PLAELM2	2	R/W	2
0xFFFF0B0C	PLAELM3	2	R/W	2
0xFFFF0B10	PLAELM4	2	R/W	2
0xFFFF0B14	PLAELM5	2	R/W	2
0xFFFF0B18	PLAELM6	2	R/W	2
0xFFFF0B1C	PLAELM7	2	R/W	2
0xFFFF0B20	PLAELM8	2	R/W	2
0xFFFF0B24	PLAELM9	2	R/W	2
0xFFFF0B28	PLAELM10	2	R/W	2
0xFFFF0B2C	PLAELM11	2	R/W	2
0xFFFF0B30	PLAELM12	2	R/W	2
0xFFFF0B34	PLAELM13	2	R/W	2
0xFFFF0B38	PLAELM14	2	R/W	2
0xFFFF0B3C	PLAELM15	2	R/W	2
0xFFFF0B40	PLACLK	1	R/W	2
0xFFFF0B44	PLAIRQ	2	R/W	2
0xFFFF0B48	PLAADC	4	R/W	2
0xFFFF0B4C	PLADIN	4	R/W	2
0xFFFF0B50	PLADOUT	4	R	2
0xFFFF0B54	PLALCK	1	W	2

表25. PWM基地址 = 0xFFFF0F80

地址	名称	字节	访问类型	周期
0xFFFF0F80	PWMCON0	2	R/W	2
0xFFFF0F84	PWM0COM0	2	R/W	2
0xFFFF0F88	PWM0COM1	2	R/W	2
0xFFFF0F8C	PWM0COM2	2	R/W	2
0xFFFF0F90	PWM0LEN	2	R/W	2
0xFFFF0F94	PWM1COM0	2	R/W	2
0xFFFF0F98	PWM1COM1	2	R/W	2
0xFFFF0F9C	PWM1COM2	2	R/W	2
0xFFFF0FA0	PWM1LEN	2	R/W	2
0xFFFF0FA4	PWM2COM0	2	R/W	2
0xFFFF0FA8	PWM2COM1	2	R/W	2
0xFFFF0FAC	PWM2COM2	2	R/W	2
0xFFFF0FB0	PWM2LEN	2	R/W	2
0xFFFF0FB4	PWMCON1	2	R/W	2
0xFFFF0FB8	PWMCLRI	2	W	2

表26. 外部存储器基地址 = 0xFFFF0000

地址	名称	字节	访问类型	周期
0xFFFF0000	XMCFG	1	R/W	2
0xFFFF0010	XM0CON	1	R/W	2
0xFFFF0014	XM1CON	1	R/W	2
0xFFFF0018	XM2CON	1	R/W	2
0xFFFF001C	XM3CON	1	R/W	2
0xFFFF0020	XM0PAR	2	R/W	2
0xFFFF0024	XM1PAR	2	R/W	2
0xFFFF0028	XM2PAR	2	R/W	2
0xFFFF002C	XM3PAR	2	R/W	2

ADuC7124/ADuC7126

表27. GPIO基地址 = 0xFFFF0400

地址	名称	字节	访问类型	周期
0xFFFF400	GP0CON	4	R/W	1
0xFFFF404	GP1CON	4	R/W	1
0xFFFF408	GP2CON	4	R/W	1
0xFFFF40C	GP3CON	4	R/W	1
0xFFFF410	GP4CON	4	R/W	1
0xFFFF420	GP0DAT	4	R/W	1
0xFFFF424	GP0SET	1	W	1
0xFFFF428	GP0CLR	1	W	1
0xFFFF42C	GP0PAR	4	R/W	1
0xFFFF430	GP1DAT	4	R/W	1
0xFFFF434	GP1SET	1	W	1
0xFFFF438	GP1CLR	1	W	1
0xFFFF43C	GP1PAR	4	R/W	1
0xFFFF440	GP2DAT	4	R/W	1
0xFFFF444	GP2SET	1	W	1
0xFFFF448	GP2CLR	1	W	1
0xFFFF44C	GP2PAR	4	R/W	1
0xFFFF450	GP3DAT	4	R/W	1
0xFFFF454	GP3SET	1	W	1
0xFFFF458	GP3CLR	1	W	1
0xFFFF45C	GP3PAR	4	R/W	1
0xFFFF460	GP4DAT	4	R/W	1
0xFFFF464	GP4SET	1	W	1
0xFFFF468	GP4CLR	1	W	1
0xFFFF46C	GP4PAR	4	R/W	1

表28. Flash/EE模块0基地址 = 0xFFFF800

地址	名称	字节	访问类型	周期
0xFFFF800	FEE0STA	1	R	1
0xFFFF804	FEE0MOD	1	R/W	1
0xFFFF808	FEE0CON	1	R/W	1
0xFFFF80C	FEE0DAT	2	R/W	1
0xFFFF810	FEE0ADR	2	R/W	1
0xFFFF818	FEE0SGN	3	R	1
0xFFFF81C	FEE0PRO	4	R/W	1
0xFFFF820	FEE0HID	4	R/W	1

表29. Flash/EE模块1基地址 = 0xFFFF880

地址	名称	字节	访问类型	周期
0xFFFF880	FEE1STA	1	R	1
0xFFFF884	FEE1MOD	1	R/W	1
0xFFFF888	FEE1CON	1	R/W	1
0xFFFF88C	FEE1DAT	2	R/W	1
0xFFFF890	FEE1ADR	2	R/W	1
0xFFFF898	FEE1SGN	3	R	1
0xFFFF89C	FEE1PRO	4	R/W	1
0xFFFF8A0	FEE1HID	4	R/W	1

ADC电路概览

该模数转换器是一个快速多通道12位ADC。工作电压为2.7 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可以在三种不同的模式下工作。

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

当工作在单端模式或伪差分模式下时，转换器模拟输入范围为0 V至 V_{REF} 。在全差分模式下，输入信号共模电压 V_{CM} 在0 V至 AV_{DD} 之间，最大幅值不超过 $2 \times V_{REF}$ (见图27)。

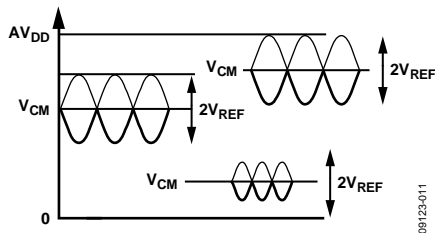


图27. 全差分模式下的平衡信号示例

片内提供一个高精度、低漂移、工厂校准的2.5V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 $\overline{CONV_START}$ 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。

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。

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0V至 V_{REF} ，输出编码为标准二进制编码

$$1 \text{ LSB} = FS/4096, \text{ 或}$$

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

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

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28所示。

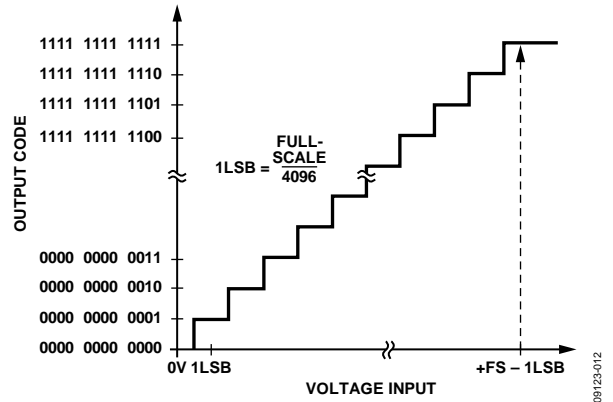


图28. 伪差分或单端模式下ADC传递函数

全差分模式

差分信号的幅值为输入引脚 V_{IN+} 和输入引脚 V_{IN-} 的信号差值(即 $V_{IN+} - V_{IN-}$)。 V_{IN+} 通过ADCCP寄存器选择， V_{IN-} 通过ADCCN寄存器选择。因此，差分信号的最大幅值为 $-V_{REF}$ 至 $+V_{REF}$ 峰峰值(即 $2 \times V_{REF}$)。这与共模模式(CM)无关。共模模式是两输入信号的平均值，同时也是两个输入电压的中点，即 $(V_{IN+} + V_{IN-})/2$ 。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 $1 \text{ LSB} = 2 \times V_{REF} / 4096$ ，或 $2 \times 2.5 \text{ V} / 4096 = 1.22 \text{ mV}$ 。输出结果为 ± 11 位，但是这个结果向右移了一位。这样，书写C语言代码时，就可以将ADCDAT中的结果声明为带符号整数。设计的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29所示。

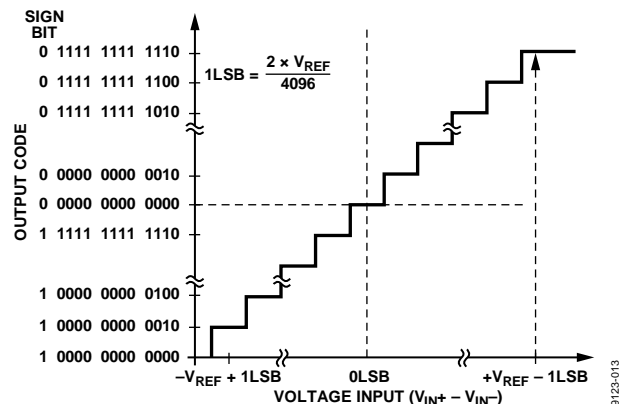


图29. 差分模式下ADC传递函数

ADuC7124/ADuC7126

典型操作

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

高四位是符号位。12位转换结果存放在如图30所示寄存器中的16位至27位。同样应注意，在全差分模式下，其结果是二进制补码格式。在伪差分模式和单端模式下，其结果是标准二进制格式。

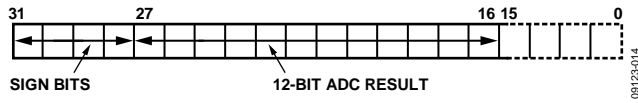


图30. ADC转换结果格式

DAC×DAT内采用相同格式，以简化软件。

功耗

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640μA。使用内部基准电压源时，电流要增加140μA。转换过程中，额外电流是0.3μA乘以采样频率(单位为kHz)。

时序

ADC时序如图31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。

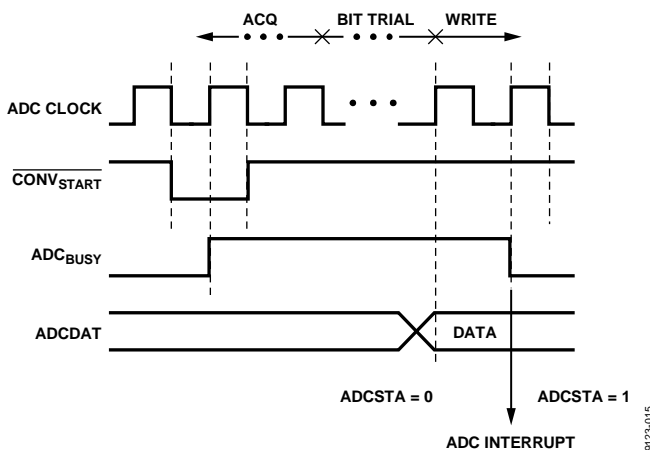


图31. ADC时序

寄存器接口

ADC通过8个寄存器进行控制和配置。

ADCCON寄存器

名称: ADCCON

地址: 0xFFFF0500

默认值: 0x0600

访问类型: 读/写

ADCCON是一个ADC控制寄存器，用户可以通过它使能ADC外设，选择ADC的工作模式(单端模式、伪差分模式、全差分模式)和转换类型等。该寄存器如表30所示。

表30. ADCCON寄存器位功能描述

位	值	描述
[15:14]		保留。
13		用户将该位置1，使能边沿触发模式。用户清0，使能电平触发模式。
[12:10]		ADC时钟速度。
	000	$f_{ADC}/1$ 。该分频器用于在外部时钟频率小于41.78MHz时获得1MSPS ADC。
	001	$f_{ADC}/2$ (默认值)。
	010	$f_{ADC}/4$ 。
	011	$f_{ADC}/8$ 。
	100	$f_{ADC}/16$ 。
	101	$f_{ADC}/32$ 。
[9:8]		ADC采集时间。
	00	2个时钟。
	01	4个时钟。
	10	8个时钟(默认值)。
	11	16个时钟。
7		转换启动使能。 用户置1启动任意类型的转换。清0禁用转换启动(清0该位不能使正进行的连续转换停止)。
6		使能ADC _{BUSY} 。 用户将该位置1，使能ADCBUSY引脚。清0则禁用该引脚。
5		ADC电源控制。 用户将该位置1，ADC置于正常模式(ADC在上电至少5μs之后才能正确转换)。清0，ADC置于掉电模式。
[4:3]		转换模式。
	00	单端模式。
	01	差分模式。
	10	伪差分模式。
	11	保留。

位	值	描述
[2:0]		转换类型。
	000	CONV _{START} 引脚输入作为转换使能信号。
	001	定时器1输出作为转换使能信号。
	010	定时器0输出作为转换使能信号。
	011	单次软件转换。转换后设为000(注意, 为了避免CONV _{START} 引脚再一次触发转换, ADCCON寄存器的第7位在启动单次软件转换后应清0)。
	100	连续软件转换。
	101	PLA转换。
	Other	保留。

ADCCP寄存器

名称: ADCCP

地址: 0xFFFF0504

默认值: 0x00

访问类型: 读/写

ADCCP是ADC正向通道选择寄存器。该寄存器如表31所示。

表31. ADCCP1寄存器位功能描述

位	值	描述
[7:5]		保留。
[4:0]		正向通道选择位。
	00000	ADC0.
	00001	ADC1.
	00010	ADC2.
	00011	ADC3.
	00100	ADC4.
	00101	ADC5.
	00110	ADC6.
	00111	ADC7.
	01000	ADC8.
	01001	ADC9.
	01010	ADC10.
	01011	ADC11.
	01100	DAC0/ADC12.
	01101	DAC1/ADC13.
	01110	DAC2/ADC14.
	01111	DAC3/ADC15.
	10000	温度传感器。
	10001	AGND(自诊断特性)。
	10010	内部基准电压(自诊断特性)。
	10011	AV _{DD} /2.
	其它	保留。

¹ ADC和DAC通道可用性取决于器件型号。详情参考订购指南。

ADCCN寄存器

名称: ADCCN

地址: 0xFFFF0508

默认值: 0x01

访问类型: 读/写

ADCCN是一个ADC反向通道选择寄存器。该寄存器如表32所示。

表32. ADCCN寄存器位功能描述

位	值	描述
[7:5]		保留。
[4:0]		反向通道选择位。
	00000	ADC0.
	00001	ADC1.
	00010	ADC2.
	00011	ADC3.
	00100	ADC4.
	00101	ADC5.
	00110	ADC6.
	00111	ADC7.
	01000	ADC8.
	01001	ADC9.
	01010	ADC10.
	01011	ADC11.
	01100	DAC0/ADC12.
	01101	DAC1/ADC13.
	01110	DAC2/ADC14.
	01111	DAC3/ADC15.
	10000	保留。
	10001	AGND.
	10010	保留。
	10011	保留。
	Others	保留。

ADCSTA寄存器

名称: ADCSTA

地址: 0xFFFF050C

默认值: 0x00

访问类型: 只读

ADCSTA是一个ADC状态寄存器, 指示ADC转换结果已完成。ADCSTA寄存器只有一个位, 即ADCReady(位0), 表示ADC的转换状态。在一次ADC转换完成后该位将该位置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}引脚变为低电平。如

ADuC7124/ADuC7126

果通过ADCCON寄存器使能,则可以在P0.5引脚输出此信息(参见“通用输入/输出”部分)。

ADCDAT寄存器

名称: ADCDAT
地址: 0xFFFF0510
默认值: 0x00000000
访问类型: 只读

ADCDAT是一个ADC数据结果寄存器,里面存放12位ADC转换结果,如图30所示。

ADCRST寄存器

名称: ADCRST
地址: 0xFFFF0514
默认值: 0x00
访问类型: 读/写

ADCRST可以复位ADC的数字接口。通过向此寄存器写入任意数据,可恢复所有ADC寄存器到默认值。

ADCGN寄存器

名称: ADCGN
地址: 0xFFFF0530
默认值: 0x0200
访问类型: 读/写

ADCGN是一个10位增益校准寄存器。

ADCOF寄存器

名称: ADCOF
地址: 0xFFFF0534
默认值: 0x0200
访问类型: 读/写

ADCOF是一个10位失调校准寄存器。

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。该结构可在三种模式下工作:差分模式、伪差分模式、单端模式。

差模

ADuC7124/ADuC7126各包含一个基于两个容性DAC的逐次逼近型ADC。图32和图33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32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。

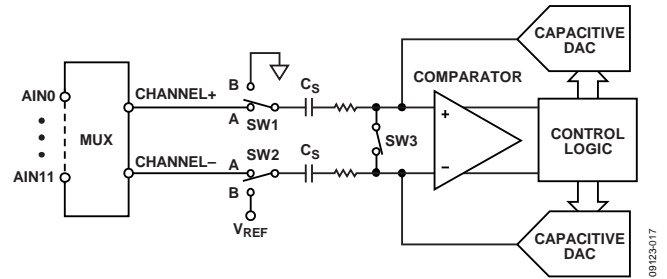


图32. ADC采样阶段

当ADC启动转换,如图33所示,SW3断开,而SW1和SW2移至位置B。这使得比较器变得不平衡。一旦转换开始,两个输入均会断开。控制逻辑和电荷再分配DAC可以加上和减去采样电容阵列中的固定电荷数量,使得比较器恢复到平衡状态。当比较器重新平衡后,转换就已经完成。控制逻辑产生ADC的输出代码。注意这里驱动 V_{IN+} 和 V_{IN-} 引脚的源输出阻抗一定要匹配,否则由于两个输入的建立时间不同会产生错误。

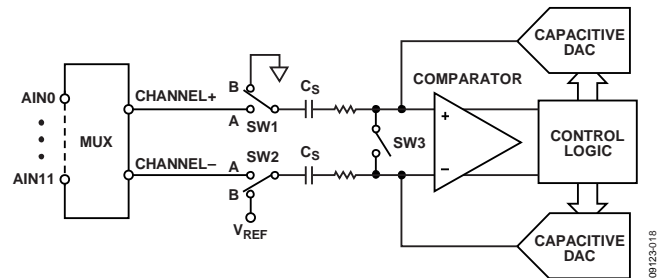


图33. ADC转换阶段

伪差分模式

在伪差分模式下,负通道(Channel-)连接到ADuC7124/ADuC7126的ADCNEG引脚。图34中,ADCNEG表示为 V_{IN-} 。SW2在A(Channel-)与B(V_{REF})之间切换。ADCNEG引脚必须接地或者接一低电压。 V_{IN+} 上的输入信号的范围为 V_{IN-} 至 $V_{REF} + V_{IN-}$ 。注意,必须恰当选择 V_{IN-} ,不要使 $V_{REF} + V_{IN-}$ 超过 AV_{DD} 。

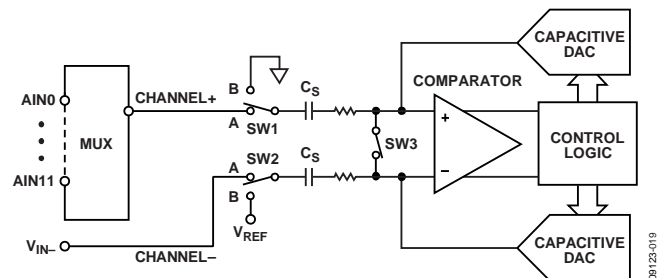


图34. 伪差分模式下ADC

单端模式

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

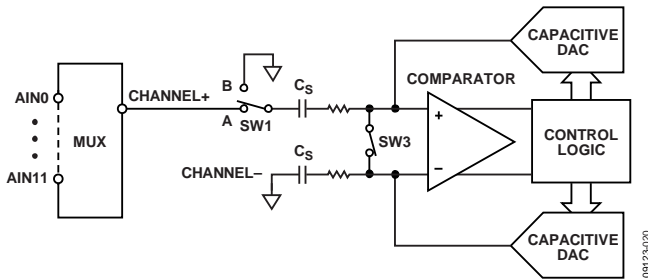


图35. 单端模式下ADC

模拟输入结构

ADC模拟输入结构等效电路如图36所示，图中4个二极管为模拟输入提供ESD保护。切记，模拟输入信号不得超过供电轨300 mV以上，否则会造成二极管正偏，并开始向基板内导通电流。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36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。这些电阻的典型值约为100 Ω。电容C2是ADC采样电容，典型值为16 pF。

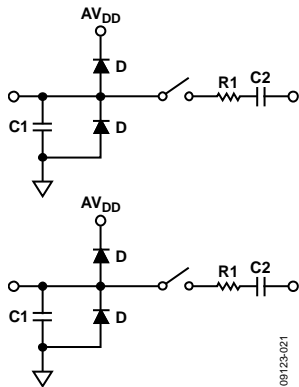


图36. 等效模拟输入电路转换阶段：开关打开，
采样阶段：开关关闭

表33. 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

在交流应用中，建议在相应的模拟输入引脚用一个RC低通滤波器来滤除模拟输入信号的高频成份。在对谐波失真和信噪比要求严格的应用中，模拟输入应采用一个低阻抗源进行驱动。高源阻抗会显著影响ADC的交流特性。这种情况下有必要使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37和图38为ADC前端的示例。

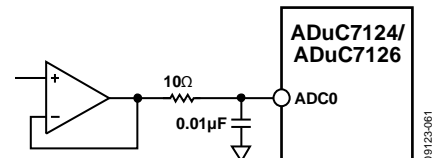


图37. 带缓冲的单端/伪差分输入

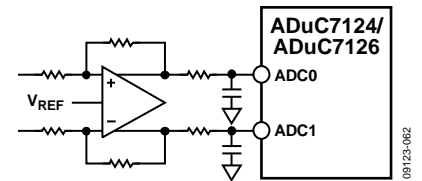


图38. 带缓冲的差分输入

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33给出了一些计算出的V_{CM}最小值和最小值。

ADuC7124/ADuC7126

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“技术规格”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通道输入级连接在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ADC结果(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 $\pm 3\%$ 。

温度传感器

ADuC7124/ADuC7126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。

ADC温度传感器转换与标准ADC电压不同。ADC性能规格并不适用于温度传感器。

内置放大器的斩波功能必须使用TSCON寄存器来使能。要使能此模式, 用户必须将TSCON的位0置1。用户还必须在此模式下取得两个连续的ADC读数并求取平均值。

ADCCON寄存器必须配置为0x37A3。

计算芯片温度的公式:

$$T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$$

其中:

T 为温度结果。

$$T_{REF} = 25^{\circ}\text{C}.$$

ADuC7124的 $V_{TREF} = 1.415\text{ V}$, ADuC7126的 $V_{TREF} = 1.392\text{ V}$,

相应的 $T_{REF} = 25^{\circ}\text{C}$, 如表1所示。

V_{ADC} 为从两个连续转换结果得出的ADC转换结果平均值。

K 为ADC在温度传感器模式下的增益, 由特性数据确定。对于ADuC7124, $K = 0.2555^{\circ}\text{C}/\text{mV}$ 。对于ADuC7126, $K = 0.2212^{\circ}\text{C}/\text{mV}$ 。它对应于表1中的电压温度系数的倒数。

使用表1中的默认值, 并且不经任何校准, 此公式变为:

$$\text{ADuC7124: } T - 25^{\circ}\text{C} = (V_{ADC} - 1415) \times 0.2555$$

$$\text{ADuC7126: } T - 25^{\circ}\text{C} = (V_{ADC} - 1392) \times 0.2212$$

其中 V_{ADC} 的单位为mV。

要提高精度, 用户应在受控温度值上执行单点校准。

对于无校准的计算, 使用 25°C 和 1415 mV (ADuC7124)/ 1392mV (ADuC7126)。单点校准的思路是使用其他已知的(T_{TREF} , V_{TREF})值来取代各器件的通用值($T = 25^{\circ}\text{C}$, $1415\text{ mV}/1392\text{mV}$)。

有些用户可能无法取得此类数值对。

这种情况下, ADuC7124/ADuC7126会在TEMPREF寄存器中载入单点校准值。欲了解此寄存器的更多信息, 请参见表35。在ADuC7124/ADuC7126的生产测试过程中, TEMPREF寄存器中载入了失调调整因子。每个器件的TEMPREF寄存器拥有不同的值。借助此单点校准, 仍可使用同样的公式:

$$T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$$

其中:

$T_{REF} = 25^{\circ}\text{C}$, 但不保证。

V_{TREF} 可使用TEMPREF寄存器算出。

TSCON寄存器

名称: TSCON

地址: 0xFFFF0544

默认值: 0x00

访问类型: 读/写

表34. TSCON寄存器位功能描述

位	描述
[7:1]	保留。
0	温度传感器斩波使能位。此位必须置1。通过将该位置1, 可使能ADC内置放大器的斩波功能。通过将该位清0, 可禁用斩波。这会导致温度传感器读数不正确。该位的默认值为0。

TEMPREF寄存器

名称:	TEMPREF
地址:	0xFFFF0548
默认值:	0XXXXX
访问类型:	读/写

表35. TEMPREF寄存器位功能描述

位	描述
[15:9]	保留。
8	温度基准电压符号位。
[7:0]	温度传感器失调校准电压。 根据TEMPREF寄存器计算 V_{TREF} 的公式如下： 如果TEMPREF符号为负， $C_{TREF} = 2292 - TEMPREF[7:0]$ 其中，TEMPREF[8] = 1 或者 如果TEMPREF符号为正， $C_{TREF} = TEMPREF[7:0] + 2292$ 其中，TEMPREF[8] = 0 最后， $V_{TREF} = ((C_{TREF} \times V_{REF}) / 4096) \times 1000$ 将 V_{TREF} 插入以下公式： $T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$ 请注意，ADC码值2292是使用TEMPREF寄存器时的默认值。该值并不是精确值，仅可用于TEMPREF寄存器。

带隙基准电压源

ADuC7124/ADuC7126内置一个2.5 V的片内带隙基准电压源，后者可用于ADC和DAC。该内部基准电压也出现在 V_{REF} 引脚上。当采用内部基准电压源时，外部 V_{REF} 引脚与AGND之间必须接上一个0.47 μ F的电容，以确保ADC转换

时的稳定性和快速响应。片内基准电压源也可以连接至一个外部引脚(V_{REF})，作为系统其它电路的基准电压源。但是由于 V_{REF} 输出的驱动能力较低(<5 μ A)，因此需要连接一个外部缓冲器。使用可编程选项，还可以在 V_{REF} 引脚上连接外部基准电压输入。注意内部基准电压不能禁用。因此，外部基准电压源必须有足够强的驱动能力来抑制内部基准源。

REFCON寄存器

名称:	REFCON
地址:	0xFFFF048C
默认值:	0x00
访问类型:	读/写

带隙基准电压接口包括一个8位REFCON寄存器，如表36所示。

表36. REFCON寄存器位功能描述

位	描述
[7:2]	保留。
1	内部基准电压关断位。 将该位置1可关断内部基准电压源。 连接外部基准电压源时应将此位置1。 将该位清0可以使能内部基准电压源。 该位的默认值为0。
0	内部基准电压输出使能。 用户置1时，内部2.5V基准电压源接在 V_{REF} 上。 基准电压源可用于外部元件，但必须增加缓冲。 用户清0时，内部2.5V基准电压源断开与 V_{REF} 引脚的连接。

如需给ADuC7124/ADuC7126连接外部基准电压源，请设置REFCON = 0x00。ADC和DAC可配置为使用相同或不同的基准电压源(见表66)。

非易失性FLASH/EE存储器

ADuC7124/ADuC7126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ADuC7124/ADuC7126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

ADuC7124/ADuC7126内置两个64 kB的Flash/EE存储器阵列。第一个模块中，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。第二个模块中，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

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

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

1. 初始页面擦除序列
2. 读/校验时序(单个Flash/EE)。
3. 存储器的字节编程时序。
4. 二次读/校验时序(耐久性周期)。

在进行可靠性验证时，Flash/EE存储器中的三页(顶、中和底)的每半个字(16位宽)可循环存取10000次(从0x0000至0xFFFF)。如表1所示，Flash/EE存储器耐久性是依据JEDEC保持期限规格A117在-40°至+125°C工业温度范围内测量出来的。这个结果满足给定温度下最小耐久性超过10,000个周期的规格。

保持期限衡量Flash/EE存储器长时间保持编程数据的能力。同样，在特定结温($T_j = 85^\circ\text{C}$)下根据标准JEDEC保持期限规格(A117)对器件进行测试。作为此测试程序的一部分，Flash/EE存储器循环工作的次数到达指定的耐久性限值(参见Flash/EE存储器部分)，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激活能的保持期限随T_j递减，如图39所示。

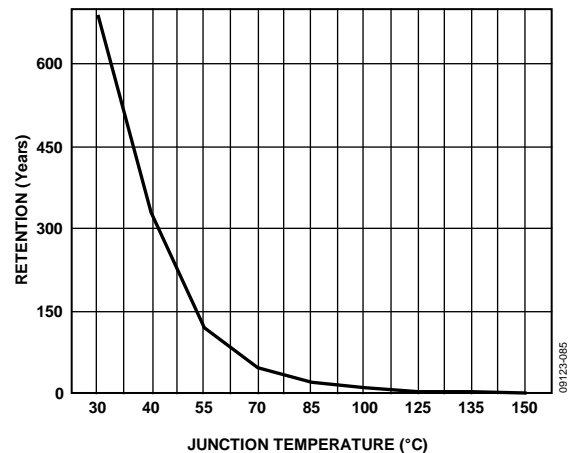


图39. Flash/EE存储器数据保持期限

编程

在串行下载模式或JTAG模式下，可对Flash/EE存储器中的62 kB区域进行在线编程。

串行下载(在线编程)

ADuC7124/ADuC7126可利用标准UART串行端口下载代码。只能使用UART0(P01.0和P1.1)。如果BM引脚通过一个1 kΩ的外部下拉电阻下拉到低电平，器件就会在复位或上电周期后进入到串行下载模式。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)。由开发系统提供的借助通用异步串口进行串行下载的功能也支持PC串行下载。应用笔记AN-724说明了UART下载协议。

通过I²C执行下载(在线编程)

ADuC7126BSTZ126I和ADuC7126BSTZ126IRL可利用I²C端口下载代码。如果BM引脚通过一个1 kΩ的外部电阻下拉到低电平且闪存地址0x80014 = 0xFFFFFFFF，这些型号就会在复位或上电周期后进入到下载模式。一旦进入下载模式，当器件在其目标应用硬件为在线状态时，用户就可以

下载代码到Flash/EE的全部126 kB中。开发系统提供PC I²C下载功能，支持通过I²C进行串行下载。ADI公司提供USB转I²C下载适配板，此板可连接到PC的USB端口和ADuC7126的I²C端口。产品型号为USB-I²C/LIN-CONV-Z。

AN-806应用笔记详细介绍了通过I²C执行串行下载的协议。

JTAG访问

JTAG协议允许借助片内JTAG接口进行代码下载和调试。

如果要通过JTAG接口访问器件，必须将P0.0/BM引脚设为高电平。

调试时，不应将用户代码写入P0.1、P0.2和P0.3引脚。如果用户切换这些引脚中的任意一个，JTAG调试器将无法连接到ADuC7124/ADuC7126。这种情况下，应使用UART/I²C下载器批量擦除器件。

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，可以保护这126 kB，使其不能通过JTAG和UART编程模式来读取。该寄存器的其他31位可以保护Flash/EE存储器不被写入；每一位保护4页，即2 kB。写保护对所有类型的访问都有效。同样，FEE1PRO和FEE1HID保护第二个64 kB模块。所有32位都用来提供保护，一次4页。

三种级别保护

- 可通过直接写入FEExHID寄存器来设定或取消保护。复位后，保护失效。
- 可通过写入FEExPRO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。FEExPRO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FEExPRO。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FEExPRO寄存器和一个特定密钥值(0xDEADDEAD)可以永久保护闪存中的数据，此时即使再次输入密钥也不允许更改FEExPRO寄存器。

密钥写入时序

1. 写入FEExPRO寄存器的某一位，相应页面会被保护；
2. 将FEExMOD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FEExADR和FEExDAT寄存器；
4. 在FEExCON寄存器中运行写密钥命令0x0C，通过监测FEExSTA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FEExPRO值。如果密钥选择的值是0xDEAD，则无法取消存储器保护。只有批量擦除能够取消器件保护，但它也会擦除所有用户代码。

密钥写入时序如下所示(保护闪存的第4页到第7页)：

```
FEExPRO=0xFFFFFFFF; //Protect Page 4 to
Page 7
FEExMOD=0x48; //Write key enable
FEExADR=0x1234; //16 bit key value
FEExDAT=0x5678; //16 bit key value
FEExCON= 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FEExADR = 0xDEAD且FEExDAT = 0xDEAD。

Flash/EE控制接口

表37. FEE0STA寄存器

名称	地址	默认值	访问类型
FEE0STA	0xFFFFF800	0x0000	R

表38. FEE0MOD寄存器

名称	地址	默认值	访问类型
FEE0MOD	0xFFFFF804	0x80	R/W

表39. FEE0CON寄存器

名称	地址	默认值	访问类型
FEE0CON	0xFFFFF808	0x00	R/W

表40. FEE0DAT寄存器

名称	地址	默认值	访问类型
FEE0DAT	0xFFFFF80C	0xXXXX	R/W

FEE0DAT是一个16位数据寄存器。

表41. FEE0ADR寄存器

名称	地址	默认值	访问类型
FEE0ADR	0xFFFFF810	0x0000	R/W

FEE0ADR是一个16位地址寄存器。

表42. FEE0SGN寄存器

名称	地址	默认值	访问类型
FEE0SGN	0xFFFFF818	0FFFFFFF	R

FEE0SGN为24位编码签名。

表43. FEE0PRO寄存器

名称	地址	默认值	访问类型
FEE0PRO	0xFFFFF81C	0x00000000	R/W

FEE0PRO在随后的寄存器复位后提供保护。它需要一个软件密钥(见表56)。

表44. FEE0HID寄存器

名称	地址	默认值	访问类型
FEE0HID	0xFFFFF820	0xFFFFFFFF	R/W

FEE0HID提供立即寄存器保护。它不需要任何软件密钥(见表56)。

ADuC7124/ADuC7126

表45. FEE1STA寄存器

名称	地址	默认值	访问类型
FEE1STA	0xFFFFF880	0x0000	R

表46. FEE1MOD寄存器

名称	地址	默认值	访问类型
FEE1MOD	0xFFFFF884	0x80	R/W

表47. FEE1CON寄存器

名称	地址	默认值	访问类型
FEE1CON	0xFFFFF888	0x00	R/W

表48. FEE1DAT寄存器

名称	地址	默认值	访问类型
FEE1DAT	0xFFFFF88C	0XXXXX	R/W

FEE1DAT是一个16位数据寄存器。

表49. FEE1ADR寄存器

名称	地址	默认值	访问类型
FEE1ADR	0xFFFFF890	0x0000	R/W

FEE1ADR是一个16位地址寄存器。

表53. FEExSTA寄存器位功能描述

位	描述
[15:6]	保留。
5	保留。
4	保留。
3	Flash/EE中断状态位。 发生中断时，即命令结束且FEExMOD寄存器的Flash/EE中断使能位置1时，自动置1。 读取FEExSTA寄存器后，该位清0。
2	Flash/EE控制器繁忙。 当控制器繁忙时，该位自动置1； 当控制器空闲时，该位自动清0。
1	命令失败。 当命令的执行失败时，该位自动置1； 读取FEExSTA寄存器后，该位自动清0。
0	命令完成。 当命令完成时，MicroConverter将该位置1。 读取FEExSTA寄存器后，该位自动清0。

表50. FEE1SGN寄存器

名称	地址	默认值	访问类型
FEE1SGN	0xFFFFF898	0FFFFFFF	R

FEE1SGN为24位编码签名。

表51. FEE1PRO寄存器

名称	地址	默认值	访问类型
FEE1PRO	0xFFFFF89C	0x00000000	R/W

FEE1PRO在随后的寄存器复位后提供保护。它需要一个软件密钥(见表57)。

表52. FEE1HID寄存器

名称	地址	默认值	访问类型
FEE1HID	0xFFFFF8A0	0FFFFFFF	R/W

FEE1HID提供立即寄存器保护。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
```

表54. FEExMOD寄存器位功能描述

位	描述
[7:5]	保留。
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55. FEExCON中的命令代码

代码	命令	描述
0x00 ¹	零点	空闲状态。
0x01 ¹	单次读取	将FEExADR索引的16位数据载入FEExDAT。
0x02 ¹	单次写入	在FEExADR指向的地址内写入FEExDAT。此操作耗时50 μs。
0x03 ¹	擦/写	擦除由FEExADR索引的存储页，并且向FEExADR指向的地址写入FEExDAT。此操作耗时20 ms。
0x04 ¹	单次验证	将FEExADR所指地址中的数据与FEExDAT中的数据进行比较。 比较的结果由FEExSTA的第1位显示。
0x05 ¹	单次擦除	擦除FEExADR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FEExSIGN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FEExPRO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，FEExCON寄存器总是立即读取0x07。

表56. FEE0PRO和FEE0HID寄存器位功能描述

位	描述
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120页、第119页到第116页和第0页到第3页的写保护。 用户清0，为这些页面提供写保护。 用户将该位置1，允许写入这些页面。

表57. FEE1PRO和FEE1HID寄存器位功能描述

位	描述
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，为这些页面提供写保护。 用户将该位置1，允许写入这些页面。
[29:0]	第119页到第116页和第0页到第3页的写保护。 用户清0，为这些页面提供写保护。 用户将该位置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24ns，而访问SRAM所需的时间为2ns，所以从SRAM中取指令只需一个时钟周期。但是，当指令涉及到存储器读写时，如果数据位于SRAM内，必须增加1个周期；如果数据位于Flash/EE内，必须增加3个周期：一个用来执行指令，另外两个从Flash/EE中获取32位数据。取一个控制流指令(例如分支指令)需要一个时钟周期，但同时要花两个时钟周期取新指令来填充流水线。

Flash/EE的执行时间

因为Flash/EE为16位，而读取16位字的时间为22ns，所以从Flash/EE执行指令无法在1个周期内完成(当CD位=0时，从SRAM执行指令就能在一个周期内完成)。并且，无论CD位为何值，在访问数据之前还存在死区时间。

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

当所执行的指令中包含使用Flash/EE进行数据存储的指令时，这两种模式下所用的时间相同。如果指令为控制流指令，除了需要1个额外的时钟周期来解码程序计数器的新

地址外，还需要4个时钟周期来填充流水线；如果1个数据处理指令只在内核寄存器中，则不需要任何额外的时钟周期，但如果所执行的数据在Flash/EE中，需要用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中读取32位的数据。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58所示。

表58. 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 ¹	2/1	1	2	1
LDH	2/1	1	1	1
LDM/PUSH	2/1	N ²	2 × N ²	N ¹
STR ¹	2/1	1	2 × 20 ns	1
STRH	2/1	1	20 ns	1
STRM/POP	2/1	N ¹	2 × N × 20 ns ¹	N ¹

¹ 一条SWAP指令包含LD和STR这两条指令，而且只读取一次，共需八个时钟周期+40ns。

² N代表使用多重装载/存储指令时，装载或存储数据的字节数(1 < N ≤ 16)。

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40所示。

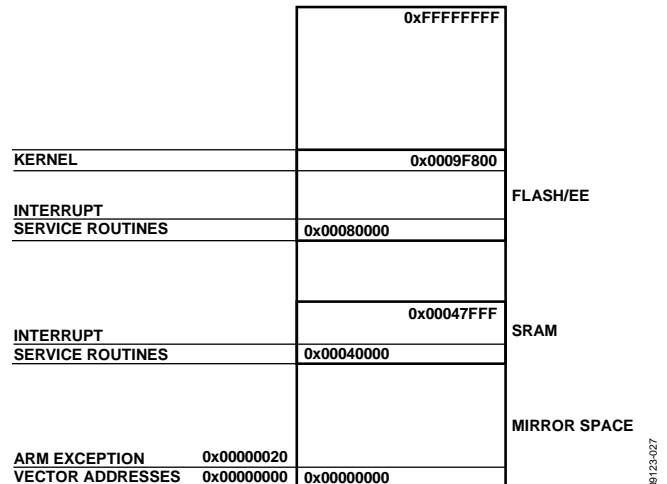


图40. 异常重映射

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。这是因为在32位ARM模式下执行异常程序时，SRAM的带宽为32位，而Flash/EE带宽为16位，所以在SRAM中执行异常程序的速度要快两倍。

表59. REMAP寄存器位功能描述
(地址 = 0xFFFF0220。默认值 = 0x00)

位	名称	描述
0	重映射	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。

重映射操作

ADuC7124/ADuC7126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。如果器件在正常模式工作(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0位将该位置1，将从Flash/EE执行重映射。必须注意从Flash/EE的地址0x00080020上方开始执行此命令，而不能从阵列底部开始执行，因为这部分已被SRAM所取代。

这个操作是可逆的。通过将REMAP寄存器的0位清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注意，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、看门狗复位和软件强制复位。RSTSTA寄存器会指示最近复位的源类型，RSTCLA则可将RSTSTA寄存器清空。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。

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。

RSTSTA寄存器

名称：	RSTSTA
地址：	0xFFFF0230
默认值：	0x01
访问类型：	只读

表60. RSTSTA寄存器位功能描述

位	描述
[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。

RSTCLR寄存器

名称：	RSTCLR
地址：	0xFFFF0234
默认值：	0x00
访问类型：	只写

请注意，清空RSTSTA寄存器时，用户必须将0x07写入RSTCLR寄存器。

RSTCFG寄存器

名称：	RSTCFG
地址：	0xFFFF024C
默认值：	0x05
访问类型：	读/写

表61. RSTCFG寄存器位功能描述

位	描述
[7:3]	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。

RSTCFG写序列描述如下：

1. 将代码0x76写入寄存器RSTKEY1。
2. 将用户值写入寄存器RSTCFG。
3. 将代码0xB1写入寄存器RSTKEY2。

ADuC7124/ADuC7126

RSTKEY0寄存器

名称: RSTKEY0
地址: 0xFFFF0248
默认值: 不适用
访问类型: 只写

RSTKEY1寄存器

名称: RSTKEY1
地址: 0xFFFF0250
默认值: 不适用
访问类型: 只写

其他模拟外设

DAC

ADuC7124/ADuC7126片内集成两个或四个12位电压输出DAC，具体取决于型号。每个DAC都有一个轨到轨电压输出缓冲器，驱动能力为5 kΩ/100 pF。

每个DAC有三种可选范围：0 V至 V_{REF} (2.5 V内部带隙基准源)、0 V至 DAC_{REF} 和0 V至 AV_{DD} 。 DAC_{REF} 相当于DAC的外部基准电压源。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4个DAC来说，这两个寄存器是相同的。这一节我们仅介绍DAC0CON(参见表和DAC0DAT(参见表的具体功能。

表62. DACxCON寄存器

名称	地址	默认值	访问类型
DAC0CON	0xFFFF0600	0x00	R/W
DAC1CON	0xFFFF0608	0x00	R/W
DAC2CON	0xFFFF0610	0x00	R/W
DAC3CON	0xFFFF0618	0x00	R/W

表63. DAC0CON寄存器位功能描述

位	值	名称	描述
[7:6]			保留。
5		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		DACCLR	DAC清除位。 用户将该位置1，使能DAC正常工作。 用户清0，复位DAC数据寄存器为0。
3			保留。该位应保留为0。
2			保留。该位应保留为0。
[1:0]			DAC范围位。 关断模式。DAC输出为三态。
	00		0 V至 DAC_{REF} 范围。
	01		0 V至 V_{REF} (2.5 V)范围。
	10		0 V至 AV_{DD} 范围。
	11		0 V至 AV_{DD} 范围。

表64. DACxDAT寄存器

名称	地址	默认值	访问类型
DAC0DAT	0xFFFF0604	0x00000000	R/W
DAC1DAT	0xFFFF060C	0x00000000	R/W
DAC2DAT	0xFFFF0614	0x00000000	R/W
DAC3DAT	0xFFFF061C	0x00000000	R/W

表65. DAC0DAT寄存器位功能描述

位	描述
[31:28]	保留。
[27:16]	DAC0的12位数据。
[15:0]	保留。

DAC的使用

片内DAC架构由一DAC电阻串和一个输出缓冲放大器构成。功能等效框图如图41所示。

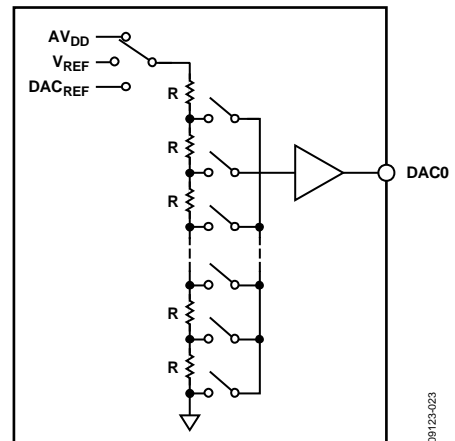


图41. DAC结构

如图41所示，用户可在软件中选择各DAC的基准电压源。它可以是 AV_{DD} 、 V_{REF} 或者 DAC_{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 DAC_{REF} 模式中，DAC输出传递函数范围为0 V至 DAC_{REF} 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。

地附近和 V_{DD} 处的线性降级是由输出放大器的饱和引起的，图42反映了这种效应的一般表现(失调误差和增益误差忽略不计)。图42中的虚线为理想的传递函数，而实线则表示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42仅代表输入范围为0 V至 AV_{DD} 模式时的传递函数。在0 V至 V_{REF} 或0 V至 DAC_{REF} 模式($V_{REF} < AV_{DD}$ 或 $DAC_{REF} < AV_{DD}$)，下半部分的非线性度是相似的。然而，传递函数的上半部分一直到端点都表现为理想的线性(这里采用 V_{REF} ，而非 AV_{DD})，这说明DAC输出没有端点线性误差。

ADuC7124/ADuC7126

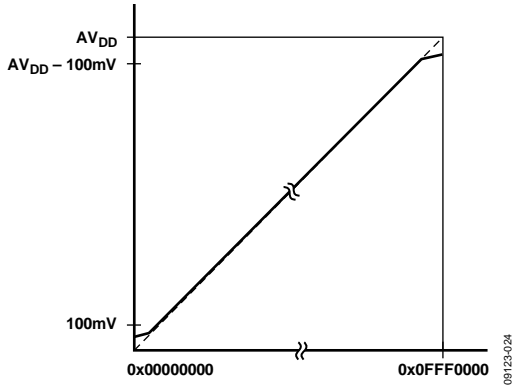


图42. 放大器饱和引起的端点非线性

当有输出负载时，图42中的端点非线性会变得更差。ADuC7124/ADuC7126数据手册中的绝大多数技术参数都是在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42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。

ADC和DAC的基准电压源

ADC和DAC均可配置为使用内部 V_{REF} 或外部基准电压源作为基准源。内部 V_{REF} 必须与外部0.47 μF电容搭配使用。

表66. ADC和DAC的基准电压源选择

REFCON[0]	DACxCON[1:0]	描述
0	00	ADC采用外部基准电压源。DAC关断。
0	01	ADC采用外部基准电压源。DAC采用 DAC_{REF} 。
0	10	保留。
0	11	ADC采用外部基准电压源。DAC采用内部 AV_{DD} 。
1	00	ADC采用内部 V_{REF} 。DAC关断。
1	01	ADC采用外部基准电压源。DAC采用 DAC_{REF} 。
1	10	ADC和DAC均采用内部 V_{REF} 。
1	11	ADC采用内部 V_{REF} 。DAC采用内部 AV_{DD} 。

注意：如果REFCON[1] = 1，则内部 V_{REF} 关断，ADC无法使用内部 V_{REF} 。

在运算放大器模式下配置DAC缓冲器

在运算放大器模式下，DAC输出缓冲器用作运算放大器，而DAC本身禁用。

如果DACBCFG位0将该位置1，则ADC0是运算放大器的正输入端，ADC1是负输入端，而DAC0是输出端。在该模式下，应通过将DAC0CON的位0和位1清0来将DAC关断。

如果DACBCFG位1将该位置1，则ADC2是运算放大器的正输入端，ADC3是负输入端，而DAC1是输出端。在该模式下，应通过将DAC1CON的位0和位1清0来将DAC关断。

如果DACBCFG位2将该位置1，则ADC4是运算放大器的正输入端，ADC5是负输入端，而DAC2是输出端。在该模式下，应通过将DAC2CON的位0和位1清0来将DAC关断。

如果DACBCFG位3将该位置1，则ADC8是运算放大器的正输入端，ADC9是负输入端，而DAC3是输出端。在该模式下，应通过将DAC3CON的位0和位1清0来将DAC关断。

DACBCFG寄存器

名称：	DACBCFG
地址：	0xFFFF0654
默认值：	0x00
访问类型：	读/写

表67. DACBCFG寄存器位功能描述

位	描述
[7:4]	保留。始终置为0。
3	通过将该位置1，可配置DAC3输出缓冲器在运算放大器模式下工作。通过将该位清0，可让该DAC缓冲器在正常模式下工作。
2	通过将该位置1，可配置DAC2输出缓冲器在运算放大器模式下工作。通过将该位清0，可让该DAC缓冲器在正常模式下工作。
1	通过将该位置1，可配置DAC1输出缓冲器在运算放大器模式下工作。通过将该位清0，可让该DAC缓冲器在正常模式下工作。
0	通过将该位置1，可配置DAC0输出缓冲器在运算放大器模式下工作。通过将该位清0，可让该DAC缓冲器在正常模式下工作。

DACBCFG写序列描述如下：

1. 将代码0x9A写入寄存器DACBKEY1。
2. 将用户值写入寄存器DACBCFG。
3. 将代码0x0C写入寄存器DACBKEY2。

DACBKEY1寄存器

名称:	DACBKEY1
地址:	0xFFFF0650
默认值:	0x0000
访问类型:	写入

DACBKEY2寄存器

名称:	DACBKEY2
地址:	0xFFFF0658
默认值:	0x0000
访问类型:	写入

电源监控器

ADuC7124/ADuC7126电源监控器负责调节片上IOV_{DD}电压，当IOV_{DD}引脚电压降到两个电源跳变点之一以下时就会给出提示。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。而一旦CMP恢复到高电平，该位会立即被清0。

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。

PSWCON寄存器

名称:	PSMCON
地址:	0xFFFF0440
默认值:	0x0008
访问类型:	读/写

表68. PSMCON寄存器位功能描述

位	名称	描述
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。 该位将该位置1，使能电源监控器；该位清0，禁用电源监控器。
0	PSMI	电源监控器中断位。当CMP变为低电平时，该位会被MicroConverter将该位置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；由于没有时间延迟，所以一旦CMP变为高电平，PSMI位可以立即被清0。

比较器

ADuC7124/ADuC7126集成一个电压比较器。比较器的正输入端与ADC2引脚复用，而负输入端则有两个：ADC3或DAC0。通过配置，电压比较器的输出可以产生系统中断、可以直接路由至可编程逻辑阵列、可以启动ADC转换或输出到外部引脚CMP_{OUT}上，如图43所示。

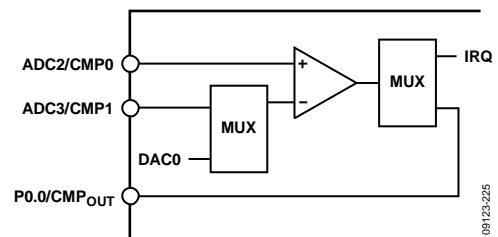


图43. 比较器

迟滞

图44描述了输入失调电压和迟滞的定义方式。其中，输入失调电压(V_{OS})为迟滞范围中心对地的差值。它可以是正，也可以是负；迟滞电压(V_H)为迟滞范围 $\frac{1}{2}$ 的宽度。

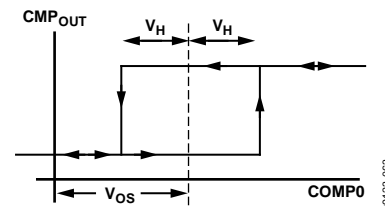


图44. 比较器迟滞传递函数

ADuC7124/ADuC7126

比较器接口

比较器接口由一个16位寄存器CMPCON组成，如表69所示。

CMPCON寄存器

名称:	CMPCON
地址:	0xFFFF0444
默认值:	0x0000
访问类型:	读/写

表69. CMPCON寄存器位功能描述

位	值	名称	描述
[15:11]			保留。
10		CMPEN	比较器使能位。将该位置1，使能比较器；清0，禁用比较器。
[9:8]		CMPIN	比较器负输入选择位。
	00		$AV_{DD}/2$ 。
	01		ADC3输入。
	10		DAC0输出。
[7:6]	11		保留。
		CMPOC	比较器输出配置位。
	00		保留。
	01		保留。
[5]	10		在CMP _{OUT} 上输出。
	11		IRQ。
		CMPOL	比较器输出逻辑状态位。该位清0时，如果正输入(CMP0)高于负输入(CMP1)，比较器输出为高电平；该位置1时，如果正输入低于负输入，比较器输出为高电平。
[4:3]		CMPRES	响应时间。
	00		对于大信号(2.5V差分)，响应时间典型值为5 μs。对于小信号(0.65V差分)，响应时间典型值为17 μs。
	11		典型值为4 μs。
	01/10		保留。
2		CMPHYST	比较器迟滞位。将该位置1，有迟滞，大约为7.5mV；清0，无迟滞。

位	值	名称	描述
1		CMPORI	比较器输出上升沿中断。受监控的电压(CMP0)出现上升沿时，该位自动置1；对该位写1可清0。
0		CMPOFI	比较器输出下降沿中断。出现下降沿时该位自动置1；由用户清0。

振荡器和锁相环—电源控制

时钟系统

ADuC7124/ADuC7126集成一个32.768 kHz ± 3%振荡器、一个时钟分频器和一个锁相环。PLL可以锁住内部振荡器或外部32.768 KHz晶振，为系统产生一个稳定的41.78 MHz时钟(UCLK)。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，实际的内核工作频率UCLK/2^{CD}为HCLK。默认的内核时钟为PLL时钟的8分频(CD = 3)或5.22 MHz。内核时钟频率也可以来自ECLK引脚上的外部时钟，如图45所示。使用内部振荡器或外部晶体时，内核时钟可以通过ECLK输出。

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。

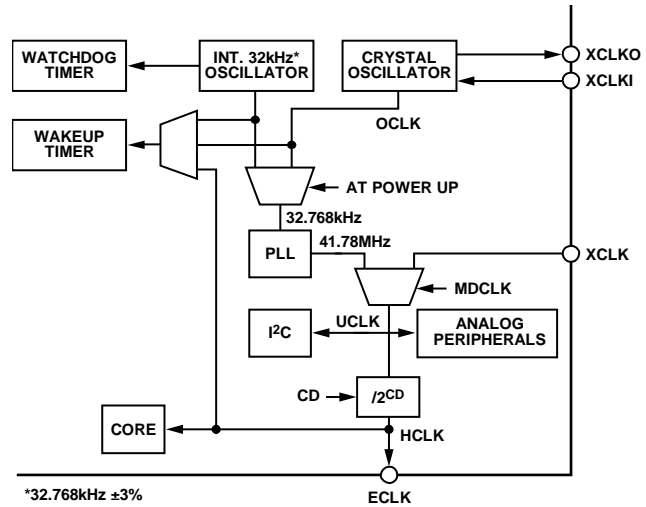


图45. 时钟系统

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

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POWCON0寄存器的正确写序列，强制器件进入浅睡眠模式。
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。

示例源代码：

```
T2LD = 5;
T2CON = 0x480;
IRQEN = 0x10;
//enable T2 interrupt
PLLKEY1 = 0xAA;
PLLCON = 0x01;
PLLKEY2 = 0x55;

POWKEY1 = 0x01;
POWCON0 = 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。

表70. 工作模式

模式	内核	外设	PLL	XTAL/T2/T3	IRQ0至IRQ3	启动/上电时间
有效	开	开	开	开	开	66 ms (CD = 0)
暂停		开	开	开	开	2.6 μ s (CD = 0), 247 μ s (CD = 7)
浅休眠			开	开	开	2.6 μ s (CD = 0), 247 μ s (CD = 7)
休眠				开	开	1.58 ms
停止					开	1.7 ms

表71. 25°C时的典型功耗(单位mA, $V_{DD} = 3.3$ V)

模式	CD = 0	CD = 1	CD = 2	CD = 3	CD = 4	CD = 5	CD = 6	CD = 7
有效	33.3	23.1	15.4	11.6	9.7	8.8	8.3	8.1
暂停	20.6	12.7	8.8	6.8	5.8	5.3	5.1	4.9
浅休眠	4.6	4.6	4.6	4.6	4.6	4.6	4.6	4.6
休眠	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
停止	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

外部时钟选择

要切换到P0.7上的外部时钟，应在模式1下配置P0.7。外部时钟频率最高可达41.78 MHz，容差为1%。

示例源代码：

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

IRQEN = 0x10;
//enable T2 interrupt
PLLKEY1 = 0xAA;
PLLCON = 0x03; //Select external clock
PLLKEY2 = 0x55;

POWKEY1 = 0x01;
POWCON0 = 0x27;
Set core into nap mode
POWKEY2 = 0xF4;
```

电源控制系统

ADuC7124/ADuC7126支持多种工作模式选项。表70给出了不同模式下各器件的上电情况和上电时间。

表71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、DAC、I²C和SPI关断。

ADuC7124/ADuC7126

寄存器和密钥

工作模式、时钟模式和可编程时钟分频器可通过PLLCON (参见表73)和POWCONx三个寄存器控制。PLLCON控制时钟系统的工作模式，POWCON0控制内核时钟频率和掉电模式，而POWCON1则控制I²C和SPI的时钟频率。

表72. PLLKEYx寄存器

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

PLLCON寄存器

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

表73. PLLCON寄存器位功能描述

位	值	名称	描述
[7:6]			保留。
5		OSEL	32 kHz PLL输入选择。将该位置1，使用内部32 kHz振荡器。默认置1。清0，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		P0.7引脚上的外部时钟。

为了防止意外编程，写入PLLCON寄存器时必须遵循特定的时序。PLLCON写序列描述如下：

1. 将代码0xAA写入寄存器PLLKEY1。
2. 将用户值写入寄存器PLLCON。
3. 将代码0x55写入寄存器PLLKEY2。

表74. POWKEYx寄存器

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

POWKEY1和POWKEY2用于防止POWCON0意外编程。

POWCON0寄存器

名称: POWCON0
地址: 0xFFFF0408
默认值: 0x0003
访问类型: 读/写

表75. POWCON0寄存器位功能描述

位	值	名称	描述
7			保留。
[6:4]		PC	工作模式。
	000		活动模式。
	001		暂停模式。
	010		浅休眠模式。
	011		休眠模式。IRQ0至IRQ3和定时器2可以唤醒该器件。
	100		停止模式。IRQ0至IRQ3可以唤醒该器件。
	Others		保留。
3			保留。
[2:0]		CD	CPU时钟分频器位。
	000		41.78 MHz.
	001		20.89 MHz.
	010		10.44 MHz.
	011		5.22 MHz.
	100		2.61 MHz.
	101		1.31 MHz.
	110		653 kHz.
	111		326 kHz.

为了防止意外编程，写入POWCONx寄存器时必须遵循特定的时序。POWCON0写序列描述如下：

1. 将代码0x01写入寄存器POWKEY1。
2. 将用户值写入寄存器POWCON0。
3. 将代码0xF4写入寄存器POWKEY2。

表76. POWKEYx寄存器

名称	地址	默认值	访问类型
POWKEY3	0xFFFF0434	0x0000	W
POWKEY4	0xFFFF043C	0x0000	W

POWKEY3和POWKEY4用于防止POWCON1意外编程。

POWCON1寄存器

名称: POWCON1
地址: 0xFFFF0438
默认值: 0x124
访问类型: 读/写

表77. POWCON1寄存器位功能描述¹

位	值	名称	描述
[15:12]			保留。
11	1	PWMPO	通过将该位清0，可关断PWM。始终清0至00。
[10:9]	00	PWMCLKDIV	
8		SPIPO	通过将该位清0，可关断SPI。
[7:6]		SPICLKDIV	SPI模块驱动时钟分频器位。
	00		41.78 MHz.
	01		20.89 MHz.
	10		10.44 MHz.
	11		5.22 MHz.
5		I2C1PO	通过将该位清0，可关断I2C1。
[4:3]		I2C1CLKDIV	I2C0模块驱动时钟分频器位。
	00		41.78 MHz.
	01		10.44 MHz.
	10		5.22 MHz.
	11		1.31 MHz.
2		I2C0PO	通过将该位清0，可关断I2C0。
[1:0]		I2C0CLKDIV	I2C1模块驱动时钟分频器位。
	00		41.78 MHz.
	01		10.44 MHz.
	10		5.22 MHz.
	11		1.31 MHz.

¹ SPI/I2C0/I2C1的分频时钟必须大于或等于通过POWCON0 [2:0]选择的CPU时钟。

POWCON1写序列描述如下：

1. 将代码0x76写入寄存器RSTKEY3。
2. 将用户值写入寄存器POWCON1。
3. 将代码0xB1写入寄存器RSTKEY4。

数字外设

通用输入/输出

ADuC7124/ADuC7126提供40个双向通用输入/输出(GPIO)引脚。所有输入输出引脚均兼容5 V电压,也就是说,GPIO支持的输入电压为5 V。

一般情况下,通过用户代码设置,许多GPIO引脚拥有多种功能(引脚功能定义参见“引脚配置和功能描述”部分)。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。可以使用GPxPAR寄存器控制以下端口是否使用内部上拉电阻:P0.0、P0.4、P0.5、P0.6、P0.7及P1的8个GPIO。

40个GPIO引脚被分成5个端口,即端口0到端口4。每个端口由4或5个寄存器控制。

一般在复位时,CPU会将P0.6默认设置改为GPIO功能。如果MRST用于外部电路,则必须有一个外部上拉电阻来保证在CPU转换模式时P0.6的电平不降低。否则,在复位周期时间内P0.6会降低。例如,如果MRST需要进入掉电模式,可在GP0CON寄存器重新配置。

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。

ADuC7124/ADuC7126器件进入省电模式后,GPIO引脚维持各自原来的状态。另外注意,通过将RSTCFG位0将该位置1,GPIO引脚可在看门狗复位或软件复位过程中保持状态不变。

表78. GPIO引脚功能描述

端口	引脚	配置			
		00	01	10	11
0	BM/P0.0	GPIO	CMP	MS0	PLAI[7]
	TDI/P0.1 ¹	GPIO/JTAG	PWM4	BLE ⁵	
	TDO/P0.2 ¹	GPIO/JTAG	PWM5	BHE ⁵	
	TRST/P0.3 ¹	GPIO/JTAG	TRST	A16 ⁵	ADC _{BUSY}
	P0.4	GPIO/IRQ0	PWM _{TRIP}	MS1 ⁵	PLAO[1]
	P0.5	GPIO/IRQ1	ADC _{BUSY}	MS2 ⁵	PLAO[2]
	P0.6	GPIO	MRST	MS3 ⁵	PLAO[3]
	P0.7	GPIO	ECLK/XCLK ²	SIN0	PLAO[4]
1	P1.0	GPIO/T1	SIN0	SCL0 ³	PLAI[0]
	P1.1	GPIO	SOUT0	SDA0 ³	PLAI[1]
	P1.2	GPIO	RTS ³	SCL1 ³	PLAI[2]
	P1.3	GPIO	CTS ³	SDA1 ³	PLAI[3]
	P1.4	GPIO/IRQ2	RI ³	SCLK ³	PLAI[4]
	P1.5	GPIO/IRQ3	DCD ³	MISO ³	PLAI[5]
	P1.6	GPIO	DSR ³	MOSI ³	PLAI[6]
	P1.7	GPIO	DTR ³	\overline{CS} ³	PLAO[0]
2	P2.0	GPIO	CONV _{START} ⁴	SOUT0	PLAO[5]
	P2.1	GPIO	PWM0	\overline{WS} ⁵	PLAO[6]
	P2.2	GPIO	PWM1	\overline{RS} ⁵	PLAO[7]
	P2.3	GPIO		AE ⁵	SIN1
	P2.4	GPIO	PWM0	MS0 ⁵	SOUT1
	P2.5	GPIO	PWM1	MS1 ⁵	
	P2.6	GPIO	PWM2	MS2 ⁵	
	P2.7	GPIO	PWM3	MS3 ⁵	
3	P3.0	GPIO	PWM0	AD0 ⁵	PLAI[8]
	P3.1	GPIO	PWM1	AD1 ⁵	PLAI[9]
	P3.2	GPIO	PWM2	AD2 ⁵	PLAI[10]
	P3.3	GPIO	PWM3	AD3 ⁵	PLAI[11]
	P3.4	GPIO	PWM4	AD4 ⁵	PLAI[12]
	P3.5	GPIO	PWM5	AD5 ⁵	PLAI[13]
	P3.6	GPIO	PWM _{TRIP}	AD6 ⁵	PLAI[14]
	P3.7	GPIO	PWM _{SYNC}	AD7 ⁵	PLAI[15]
4	P4.0	GPIO	SIN1	AD8 ⁵	PLAO[8]
	P4.1	GPIO	SOUT1	AD9 ⁵	PLAO[9]
	P4.2	GPIO		AD10 ⁵	PLAO[10]
	P4.3	GPIO		AD11 ⁵	PLAO[11]
	P4.4	GPIO		AD12 ⁵	PLAO[12]
	P4.5	GPIO/RTCK		AD13 ⁵	PLAO[13]
	P4.6	GPIO		AD14 ⁵	PLAO[14]
	P4.7	GPIO		AD15 ⁵	PLAO[15]

¹ 用户代码不应使用这些引脚。

² 在模式1下配置时,P0.7默认为ECLK或内核时钟输出。如果将其配置为时钟输入,PLLCON的MDCLK位必须置为11。

³ SPM配置参见表90。

⁴ CONV_{START}信号在P2.0的所有模式下均有效。

⁵ 仅ADuC7126提供外部存储器接口信号。

表79. GPxCON寄存器

名称	地址	默认值	访问类型
GP0CON	0xFFFFF400	0x00000000	R/W
GP1CON	0xFFFFF404	0x00000000	R/W
GP2CON	0xFFFFF408	0x00000000	R/W
GP3CON	0xFFFFF40C	0x00000000	R/W
GP4CON	0xFFFFF410	0x00000000	R/W

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80所示。

表80. GPxCON寄存器位功能描述

位	描述
[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81. GPxPAR寄存器

名称	地址	默认值	访问类型
GP0PAR	0xFFFFF42C	0x20000000	R/W
GP1PAR	0xFFFFF43C	0x00000000	R/W
GP2PAR	0xFFFFF44C	0x000000FF	R/W
GP3PAR	0xFFFFF45C	0x00222222	R/W
GP4PAR	0xFFFFF46C	0x00000000	R/W

通过对GPxPAR寄存器编程，可分别控制端口0、1、2、3、4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82. GPxPAR寄存器位功能描述

位	描述
31	保留。
[30:29]	Px.7驱动强度。
28	Px.7上拉禁用。
27	保留。
[26:25]	Px.6驱动强度。
24	Px.6上拉禁用。
23	保留。
[22:21]	Px.5驱动强度。
20	Px.5上拉禁用。
19	保留。
[18:17]	Px.4驱动强度。

位	描述
16	Px.4上拉禁用。
15	Reserved.
[14:13]	Px.3驱动强度。
12	Px.3上拉禁用。
11	保留。
[10:9]	Px.2驱动强度。
8	Px.2上拉禁用。
7	保留。
[6:5]	Px.1驱动强度。
4	Px.1上拉禁用。
3	保留。
[2:1]	Px.0驱动强度。
0	Px.0上拉禁用。

表83. GPIO驱动强度控制位功能描述

控制位值	位描述
00	中等驱动强度。
01	低驱动强度。
1x	高驱动强度。

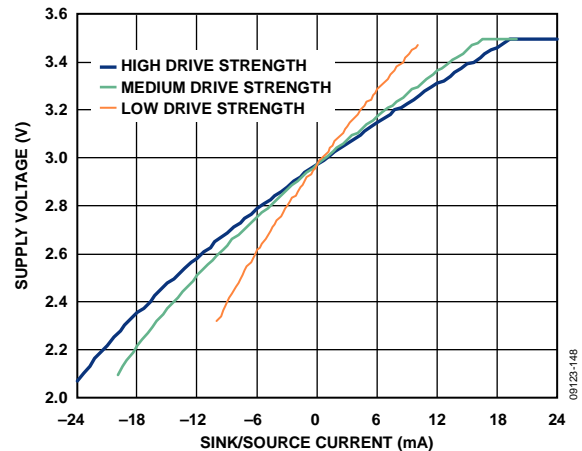


图46. 高电平的可编程强度

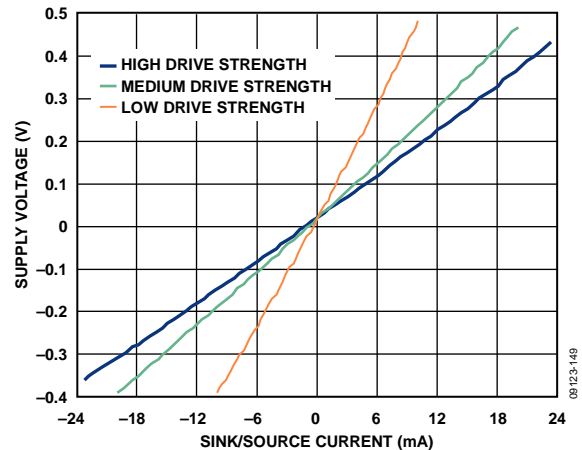


图47. 低电平的可编程强度

ADuC7124/ADuC7126

驱动强度位仅可在复位后写入一次。之后继续写入相关位时，驱动强度并不会发生任何改变。对于GPIO端口，GPIO驱动强度和上拉禁用并非始终可调。有些控制位无法更改。详情参见表78。

表84. GPxDAT寄存器

名称	地址	默认值	访问类型
GP0DAT	0xFFFFF420	0x000000XX	R/W
GP1DAT	0xFFFFF430	0x000000XX	R/W
GP2DAT	0xFFFFF440	0x000000XX	R/W
GP3DAT	0xFFFFF450	0x000000XX	R/W
GP4DAT	0xFFFFF460	0x000000XX	R/W

GPxDAT是端口x的配置和数据寄存器。它们用来配置端口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保存输入值。

表85. GPxDAT寄存器位功能描述

位	描述
[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86. GPxSET寄存器

名称	地址	默认值	访问类型
GP0SET	0xFFFFF424	0x000000XX	W
GP1SET	0xFFFFF434	0x000000XX	W
GP2SET	0xFFFFF444	0x000000XX	W
GP3SET	0xFFFFF454	0x000000XX	W
GP4SET	0xFFFFF464	0x000000XX	W

GPxSET为端口x的数据设置寄存器。

表87. GPxSET寄存器位功能描述

位	描述
[31:24]	保留。
[23:16]	数据端口x设置位。 用户将该位置1，则端口x的相应位将该位置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。

表88. GPxCLR寄存器

名称	地址	默认值	访问类型
GP0CLR	0xFFFFF428	0x000000XX	W
GP1CLR	0xFFFFF438	0x000000XX	W
GP2CLR	0xFFFFF448	0x000000XX	W
GP3CLR	0xFFFFF458	0x000000XX	W
GP4CLR	0xFFFFF468	0x000000XX	W

GPxCLR为端口x的数据清除寄存器。

表89. GPxCLR寄存器位功能描述

位	描述
[31:24]	保留。
[23:16]	数据端口x清除位。 用户将该位置1，则端口x的相应位清0，同时GPxDAT寄存器相关的位也被清0。 用户清0不会影响数据输出。
[15:0]	保留。

串口多路复用器

串口多路复用器将串口外设(SPI、UART和两个I²C)和可编程逻辑阵列(PLA)复用到10个GPIO引脚。每一个引脚都必须为其配置如表90描述的具体的I/O功能。

表90. SPM配置

SPM	GPIO (00)	UART (01)	UART/I ² C/SPI (10)	PLA (11)
SPM0	P1.0	SIN0	I2C0SCL	PLAI[0]
SPM1	P1.1	SOUT0	I2C0SDA	PLAI[1]
SPM2	P1.2	RTS	I2C1SCL	PLAI[2]
SPM3	P1.3	CTS	I2C1SDA	PLAI[3]
SPM4	P1.4	RI	SCLK	PLAI[4]
SPM5	P1.5	DCD	MISO	PLAI[5]
SPM6	P1.6	DSR	MOSI	PLAI[6]
SPM7	P1.7	DTR	\overline{CS}	PLAO[0]
SPM8	P0.7	ECLK/XCLK	SIN0	PLAO[4]
SPM9	P2.0	\overline{CONV}_{START}	SOUT0	PLAO[5]
SPM10	P4.0	SIN1	AD8	PLAO[8]
SPM11	P4.1	SOUT1	AD9	PLAO[9]
SPM12	P2.3	N/A	AE	SIN1
SPM13	P2.4	PWM0	MSO	SOUT1

表90详细描述了每一个SPMMUX引脚的模式。这些配置需通过GP0CON、GP1CON和GP2CON寄存器来完成。默认情况下，这十个引脚被配置为通用I/O引脚。

UART串行接口

该UART外设是一个全双工的、通用异步接收/发送器。UART的作用是对从外设接收的数据字符进行串并转换，对从CPU接收的数据字符进行并串转换。ADuC7124/ADuC7126配有两个行业标准16,450型UART(UART0和UART1)。每个UART具有一个小数分频器，用以支持高精度波特率生成；发送器和接收器可配有一个16字节FIFO。这两个UART均可配置为FIFO模式或非FIFO模式。

在进行串行通信时，UART遵循一个非同步的协议，支持不同的字节长度、停止位和奇偶校验位。

波特率发生器

有两种方式可生成UART波特率：普通450 UART波特率发生器和小数分频器。

普通450 UART波特率生成

该波特率是内核时钟的一个分频，分为两部分，低字节和高字节分别存放在COMxDIV0和COMxDIV1寄存器中(16位值，DL)。

$$\text{Baud Rate} = \frac{41.78 \text{ MHz}}{2^{CD} \times 16 \times 2 \times DL}$$

表 91给出了一些常用的波特率值。

表91. 普通波特率发生器的波特率

波特率	CD	DL	实际波特率	百分比误差
9600	0	0x88	9600	0
19,200	0	0x44	19,200	0
115,200	0	0x0B	118,691	3
9600	3	0x11	9600	0
19,200	3	0x08	20,400	6.25
115,200	3	0x01	163,200	41.67

小数分频器

小数分频器中集成了一个普通波特率发生器，能够产生范围更广更精确的波特率。

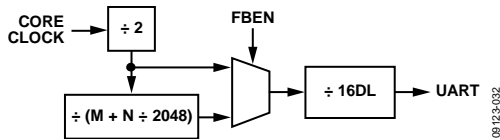


图48. 波特率生成选项

采用小数分频器的波特率计算公式如下：

$$\text{Baud Rate} = \frac{41.78 \text{ MHz}}{2^{CD} \times 16 \times DL \times 2 \times \left(M + \frac{N}{2048} \right)}$$

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{\text{Baud Rate} \times 2^{CD} \times 16 \times DL \times 2}$$

例如，设置波特率为19200，CD位=3，由表91可知DL=0x08，代入公式可得：

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{19,200 \times 2^3 \times 16 \times 8 \times 2}$$

$$M + \frac{N}{2048} = 1.06$$

其中：

$$M = 1.$$

$$N = 0.06 \times 2048 = 128.$$

$$\text{Baud Rate} = \frac{41.78 \text{ MHz}}{2^3 \times 16 \times 8 \times 2 \times \left(\frac{128}{2048} \right)}$$

其中：

$$\text{波特率} = 19,200 \text{ bps.}$$

误差=0%，而同样情况下，利用普通波特率发生器产生19200bps的波特率时，误差为6.25%。

UART寄存器定义

COM0TX寄存器

名称：	COM0TX
地址：	0xFFFF0700
默认值：	0x00
访问类型：	读/写

COM0TX为UART0的8位发送寄存器。

COM1TX寄存器

名称：	COM1TX
地址：	0xFFFF0740
默认值：	0x00
访问类型：	读/写

COM1TX为UART1的8位发送寄存器。

COM0RX寄存器

名称：	COM0RX
地址：	0xFFFF0700
默认值：	0x00
访问类型：	只读

COM0RX为UART0的8位接收寄存器。

COM1RX寄存器

名称：	COM1RX
地址：	0xFFFF0740
默认值：	0x00
访问类型：	只读

COM1RX为UART1的8位接收寄存器。

COM0DIV0寄存器

名称：	COM0DIV0
地址：	0xFFFF0700
默认值：	0x00
访问类型：	读/写

COMDIV0是UART0的低字节分频锁存器。COM0TX、COM0RX和COM0DIV0共用相同的地址。当寄存器COM0CON0的位7清0时，允许访问寄存器COM0TX和COM0RX。当寄存器COM0CON0的位7置1时，允许访问寄存器COM0DIV0。

ADuC7124/ADuC7126

COM1DIV0寄存器

名称: COM1DIV0
地址: 0xFFFF0740
默认值: 0x00
访问类型: 读/写

COM1DIV0是UART1的低字节分频锁存器。COM1TX、COM1RX和COM1DIV0共用相同的地址。当寄存器COM1CON0的位7清0时,允许访问寄存器COM1TX和COM1RX。当寄存器COM1CON0的位7置1时,允许访问寄存器COM1DIV0。

COM0IEN0寄存器

名称: COM0IEN0
地址: 0xFFFF0704
默认值: 0x00
访问类型: 读/写

COM0IEN0是UART0的中断使能寄存器。

COM1IEN0寄存器

名称: COM1IEN0
地址: 0xFFFF0744
默认值: 0x00
访问类型: 读/写

COM1IEN0是UART1的中断使能寄存器。

表92. COMxIEN0寄存器位功能描述

位	名称	描述
[7:4]		保留。
3	EDSSI	调制解调器状态中断使能位。用户置1后,当COMXSTA1[3:1]的任一位被置1时,允许产生中断。由用户清0。
2	ELSI	Rx状态中断使能位。用户置1后,当COMXSTA0[3:0]的任一位被置1时,允许产生中断。由用户清0。
1	ETBEI	使能发送缓冲空中断。用户置1后,在数据发送期间,当发送数据缓冲器为空时会产生一个中断。由用户清0。
0	ERBFI	使能接收缓冲满中断。在非FIFO模式下,用户置1后,在接收期间,当缓冲器满时,使能中断。由用户清0。在FIFO模式下,用户置1后,达到触发水平时使能中断。它还控制字符接收超时中断。由用户清0。

COM0DIV1寄存器

名称: COM0DIV1
地址: 0xFFFF0704
默认值: 0x00
访问类型: 读/写

COM0DIV1是UART0的高字节分频锁存寄存器。

COM1DIV1寄存器

名称: COM1DIV1
地址: 0xFFFF0744
默认值: 0x00
访问类型: 读/写

COM1DIV1是UART1的高字节分频锁存寄存器。

COM0IID0寄存器

名称: COM0IID0
地址: 0xFFFF0708
默认值: 0x01
访问类型: 只读

COM0IID0是UART0的中断识别寄存器。它还指示UART是否处于FIFO模式。

COM1IID0寄存器

名称: COM1IID0
地址: 0xFFFF0748
默认值: 0x01
访问类型: 只读

COM1IID0是UART1的中断识别寄存器。它还指示UART是否处于FIFO模式。

表93. COMxIID0寄存器位功能描述

位	名称	描述
[7:6]	FIFOMODE	FIFO模式标志。 0x0:非FIFO模式。 0x1:保留。 0x2:保留。 0x3:FIFO模式。如果FIFOEN将该位置1, 此位自动置1。
[5:4]	保留。	
[3:1]	STATUS[2:0]	中断状态位, 仅当NINT置1时有效。 [000]: 调制解调器状态中断。读取COMxSTA1后清0。优先级4。 [001]: 对于非FIFO模式, 使能发送缓冲空中断。 对于FIFO模式, 发送FIFO为空。写入COMxTX或读取COMxIID0时清0。优先级3。 [010]: 非FIFO模式。接收缓冲数据就绪中断。读取COMxRX后自动清0。 对于FIFO模式, 达到设定的触发水平。FIFO降到触发水平以下时自动清0。优先级2。 [011]: 接收线路状态错误中断。读取COMxSTA0后清0。优先级1。 [110]: 接收FIFO超时中断(仅FIFO模式)。如果接收FIFO中至少有一个字节, 并且在接下来的四帧访问周期中不访问接收FIFO, 则自动置1。读取COMxRX、设置RXRST或当有新字节到达接收FIFO时清0。优先级2。 [其它状态]: 保留。
0	NINT	由STATUS[2:0]将该位置1, 禁用中断标志。清0则使能中断。

¹一帧时间是指分配给一个起始位、n个数据位、一个奇偶校验位和一个停止位的时间。其中, n为COMxCON0的WLS位选择的字长。

WLS[1:0] = 00: 超时阈值 = 32位的时间 = (1 + 5 + 1 + 1) × 4。

WLS[1:0] = 01: 超时阈值 = 36位的时间 = (1 + 6 + 1 + 1) × 4。

WLS[1:0] = 10: 超时阈值 = 40位的时间 = (1 + 7 + 1 + 1) × 4。

WLS[1:0] = 11: 超时阈值 = 44位的时间 = (1 + 8 + 1 + 1) × 4。

COM0FCR寄存器

名称:	COM0FCR
地址:	0xFFFF0708
默认值:	0x00
访问类型:	读/写

COM1FCR寄存器

名称:	COM1FCR
地址:	0xFFFF0748
默认值:	0x00
访问类型:	读/写

FIFO控制寄存器(FCR)是一个只写寄存器, 其地址与中断识别寄存器(IIR, 只读)相同。

表94. COMxFCR寄存器位功能描述

位	名称	描述
[7:5]	RXFIFOTL	接收器FIFO触发水平。RXFIFOTL设置接收器FIFO的触发水平。达到触发水平时, 生成接收器数据就绪中断(如果使能中断请求)。当FIFO降到触发水平以下时, 中断清0。 0x0:1字节。 0x1:2字节。 0x2:4字节。 0x3:6字节。 0x4:8字节。 0x5:10字节。 0x6:12字节。 0x7:14字节。
[4:3]	保留。	
2	TXRST	发送FIFO复位。写入1将刷新发送FIFO。不影响移位寄存器。注意: 刷新后, 为使发送FIFO有效, TXRST应自动清0。
1	RXRST	接收FIFO复位。写入1将刷新接收FIFO。不影响移位寄存器。注意: 刷新后, 为使接收FIFO有效, RXRST应自动清0。
0	FIFOEN	发送器和接收器FIFO模式使能。必须先将FIFOEN将该位置1, 然后才能写入其它FCR位。FIFO模式置1。使能发送器和接收器FIFO。非FIFO模式清0。禁用发送器和接收器FIFO, FIFO指针清0。

COM0CON0寄存器

名称:	COM0CON0
地址:	0xFFFF070C
默认值:	0x00
访问类型:	读/写

COM0CON0是UART0的线路控制寄存器。

ADuC7124/ADuC7126

COM1CON0寄存器

名称:	COM1CON0
地址:	0xFFFF074C
默认值:	0x00
访问类型:	读/写

COM1CON0是UART1的线路控制寄存器。

表95. COMxCON0寄存器位功能描述

位	名称	描述
7	DLAB	分频锁存器访问。 用户将该位置1, 允许访问寄存器COMxDIV0和COMxDIV1。 用户清0, 禁止访问COMxDIV0和COMxDIV1, 允许访问COMxRX和COMxTX。
6	BRK	设置断开。 用户将该位置1, 强制SOUTx为0。 用户清0, 采用正常模式工作。
5	SP	强制奇偶校验。 用户将该位置1, 强制奇偶校验位为特定值: 若EPS = 1, PEN = 1, 则为1; 若EPS = 0, PEN = 1, 则为0。
4	EPS	偶校验选择位。 将该位置1, 以选择偶校验。 清0, 以选择奇校验。
3	PEN	奇偶校验使能位。 用户将该位置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	停止	停止位。 用户将该位置1, 在字长为5位时发送1½个停止位, 或在字长为6、7、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, 在发送数据内产生一个停止位。
[1:0]	WLS	字长选择: 00 = 5位, 01 = 6位, 10 = 7位, 11 = 8位。

COM0CON1寄存器

名称:	COM0CON1
地址:	0xFFFF0710
默认值:	0x00
访问类型:	读/写

COM0CON1是UART0的调制解调器控制寄存器。

COM1CON1寄存器

名称:	COM1CON1
地址:	0xFFFF0750
默认值:	0x00
访问类型:	读/写

COM1CON1是UART1的调制解调器控制寄存器。

表96. COMxCON1寄存器位功能描述

位	名称	描述
[7:5]		保留。
4	LOOPBACK	回送。 用户将该位置1, 使能回送模式。 此时强制SOUTx为高电平。 调制解调器的信号直接连接到状态输入, 即RTS连接到CTS, DTR连接到DSR。 用户清0, 返回正常模式。
3	PEN	奇偶校验使能位。 用户将该位置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	停止	停止位。 用户将该位置1, 在字长为5位时发送1½个停止位, 或在字长为6、7、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, 在发送数据内产生一个停止位。
1	RTS	发送请求位。 用户将该位置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。 用户将该位置1, 强制DTR输出为0。 用户清0, 强制DTR输出为1。

COM0STA0寄存器

名称:	COM0STA0
地址:	0xFFFF0714
默认值:	0xE0
访问类型:	只读

COM0STA0为UART0的线路状态寄存器。

COM1STA0寄存器

名称:	COM1STA0
地址:	0xFFFF0754
默认值:	0xE0
访问类型:	只读

COM1STA0为UART1的线路状态寄存器。

表97. COMxSTA0寄存器位功能描述

位	名称	描述
11	RX_error	如果PE、FE或BI将该位置1，此位自动置1。 如果PE、FE或BI清0，此位自动清0。
10	RX_timeout	仅适用于FIFO模式。如果接收FIFO中至少有一个字节，并且在接下来的4字节访问周期中不访问接收FIFO，则自动置1。
9	RX_triggered	仅适用于FIFO模式。如果Rx FIFO数超过触发水平(由FIFO控制寄存器COMxFCR[7:5]配置)，此位自动置1。 当Rx FIFO数等于或小于触发水平时，此位自动清0。
8	TX_full	仅适用于FIFO模式。如果Tx FIFO已满，此位自动置1。如果Tx FIFO不满，则此位自动清0。
7	TX_half_empty	仅适用于FIFO模式。如果Tx FIFO半空(Tx FIFO中的字节数 ≤ 8)，此位自动置1。如果Tx FIFO接收的字节数多于8，此位自动清0。
6	TEMT	COMxTX空状态位。 对于非FIFO模式，THR和TSR均为空。 对于FIFO模式，Tx FIFO和TSR均为空。
5	THRE	COMxTX和发送移位寄存器为空。 对于非FIFO模式，发送保持寄存器(THR)为空，或者THR的内容已被传输到发送移位寄存器(TSR)。 对于FIFO模式，Tx FIFO为空，或者FIFO中的最后字符已被传输到发送移位寄存器(TSR)。
4	BI	断开错误。 当SINx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。 当产生无效停止位时置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。

位	名称	描述
1	OE	溢出错误。 对于非FIFO模式，如果当前数据在读取前被覆盖，该位自动置1。该位自动清0。 对于FIFO模式，如果检测到溢出错误，该位自动置1。溢出错误仅发生在FIFO已满且移位寄存器已完全接收下一个字符之后。 新字符覆盖移位寄存器中的字符，但不会传输到FIFO。
0	DR	数据就绪。 对于非FIFO模式，COMxRX为满时自动置1。读取COMxRX后清0。 对于FIFO模式，如果COMxRX中至少有一个字节未被读取，此位自动置1。

COM0STA1寄存器

名称:	COM0STA1
地址:	0xFFFF0718
默认值:	0x00
访问类型:	只读

COM0STA1为调制解调器状态寄存器。

COM1STA1寄存器

名称:	COM1STA1
地址:	0xFFFF0758
默认值:	0x00
访问类型:	只读

COM1STA1为调制解调器状态寄存器。

表98. COMxSTA1寄存器位功能描述

位	名称	描述
7	DCD	数据载波检测。
6	RI	响铃指示。
5	DSR	数据准备就绪。
4	CTS	清除发送。
3	DDCD	三角波DCD。上一次读取COMxSTA1后，如果DCD状态改变，则自动置1。读取COMxSTA1后自动清0。
2	TERI	下降沿RI。上一次读COMxSTA1后，如果RI由0变为1则置1。读取COMxSTA1后自动清0。
1	DDSR	三角波DSR。上一次读取COMxSTA1后，如果DSR状态改变，则自动置1。读取COMxSTA1后自动清0。
0	DCTS	三角波CTS。上一次读取COMxSTA1后，如果CTS改变状态，则自动置1。读取COMxSTA1后自动清0。

ADuC7124/ADuC7126

COM0DIV2寄存器

名称:	COM0DIV2
地址:	0xFFFF072C
默认值:	0x0000
访问类型:	读/写

COM0DIV2为UART0的16位小数波特率分频寄存器。

COM1DIV2寄存器

名称:	COM1DIV2
地址:	0xFFFF076C
默认值:	0x0000
访问类型:	读/写

COM1DIV2为UART1的16位小数波特率分频寄存器。

表99. COMxDIV2寄存器位功能描述

位	名称	描述
15	FBEN	小数波特率发生器使能位。 用户将该位置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。如果FBM = 0, M = 4(具体使用请看小数分频器部分)。
[10:0]	FBN[10:0]	N。(具体使用请看小数分频器部分)。

串行外设接口

ADuC7124/ADuC7126片内集成一个完整的硬件串行外设接口(SPI)。SPI是一个工业标准同步串行接口, 允许同时双向传输8位数据(即全双工), 最大比特率可达20 Mbps。

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

MISO(主机输入, 从机输出)引脚

在主机模式下, MISO引脚被配置为输入线路; 在从机模式下, 配置为输出线路。主机上的MISO线路(数据输入)应与从机内的MISO线路(数据输出)相连。传送的数据是以字节(8位)为单位的串行数据, MSB优先。

MOSI(主机输出, 从机输入)引脚

在主机模式下, MOSI引脚被配置为输出线路; 在从机模式下, 配置为输入线路。主机上的MOSI线路(数据输出)应与从机内的MOSI线路(数据输入)相连。传送的数据是以字节(8位)为单位的串行数据, MSB优先。

SCLK(串行时钟输入/输出)引脚

主机串行时钟(SCLK)用于同步MOSI SCLK周期中发送和接收的数据。所以, 发送/接收一个字节需要8个SCLK周期。在主机模式下, SCLK引脚配置成输出端, 而在从机模式下, 配置成输入端。

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

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

SPI时钟的最高频率与时钟分频器位无关。

在从机模式下, 可对SPICON寄存器进行设置, 以配置预期输入时钟的相位和极性。从机可以从外部主机处接收数据(速率可达10 Mbps)。

在主机模式和从机模式下, 数据都在SCLK信号的一个沿发送, 并在另一个沿采样。因此, 从机时钟的极性和相位必须与主机的配置一致。

\overline{CS} (SPI片选输入)引脚

在SPI从机模式时, 置位 \overline{CS} 引脚将启动数据传输, 该引脚为一个低电平有效输入信号。然后, SPI端口开始发送和接收8位数据, 直到发送结束时为止, 此时 \overline{CS} 无效。在从机模式下, \overline{CS} 总是为输入。

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

SPI功能的外部引脚配置

ADuC7124/ADuC7126的SPI引脚为P1.4至P1.7。

P1.7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。

P1.4为SCLK引脚。

P1.5为主机输入、从机输出(MISO)引脚。

P1.6为主机输出、从机输入(MOSI)引脚。

要配置P1.4至P1.7在SPI模式下工作, 参见“通用输入/输出”部分。

SPI寄存器

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

SPI状态寄存器

名称：	SPISTA
地址：	0xFFFF0A00
默认值：	0x0000
访问类型：	只读
功能：	该32位寄存器用于存储主机、从机模式下SPI接口的状态。

表100. SPISTA寄存器位功能描述

位	名称	描述
[15:12]		保留。
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中的字节的个数不超过由SPICON寄存器中的SPIMDE位规定的个数时，该字节被清0。
[10:8]	SPIRXFSTA[2:0]	SPI接收FIFO状态位。 [000] = 接收FIFO为空。 [001] = FIFO内有1个有效字节。 [010] = FIFO内有2个有效字节。 [011] = FIFO内有3个有效字节。 [100] = FIFO内有4个有效字节。
7	SPIFOF	SPI接收FIFO溢出状态位。 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位将该位置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位将该位置1，否则该位被置1将产生一个中断。 读取SPISTA寄存器的内容后，该位被清0。
[3:1]	SPITXFSTA[2:0]	SPI发送FIFO状态位。 [000] = 发送FIFO为空。 [001] = FIFO内有1个有效字节。 [010] = FIFO内有2个有效字节。 [011] = FIFO内有3个有效字节。 [100] = FIFO内有4个有效字节。
0	SPIISTA	SPI中断状态位。 SPI中断发生时，该位被置1。 读取SPISTA寄存器的内容后，该位被清0。

ADuC7124/ADuC7126

SPIRX寄存器

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

SPITX寄存器

名称: SPITX
地址: 0xFFFF0A08
默认值: 0x00
访问类型: 只读
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: 0x00
访问类型: 读/写
功能: 该8位寄存器是SPI波特率选择寄存器。

SPICON寄存器

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

表101. SPICON寄存器位功能描述

	名称	描述
[5:14]	SPIMDE	SPI IRQ模式位。这些位用于配置在传输过程中何时发生发送/接收中断。 [00] = 传输完1个字节后, 产生发送中断。FIFO接收到一个或以上字节后, 产生接收中断。 [01] = 传输完2个字节后, 产生发送中断。FIFO接收到两个或以上字节后, 产生接收中断。 [10] = 传输完3个字节后, 产生发送中断。FIFO接收到1个或以上字节时, 产生接收中断。 [11] = 传输完4个字节后, 产生发送中断。当接收FIFO已满或存在四个字节时, 产生接收中断。
13	SPITFLH	SPI发送FIFO清空使能位。 该位置1时, 清空发送FIFO。该位无法自清0; 需要一个单次清空操作时, 应将该位置1。 如果该位的值总保持为1, 那么, 发送0x00还是最后被发送的数值取决于SPIZEN位的值。 该位为1时, 无法对发送FIFO进行写操作。 该位清0时, 禁用发送FIFO清空。
12	SPIRFLH	SPI接收FIFO清除使能位。 该位置1时, 清空接收FIFO。该位无法自清0; 需要一个单次清空操作时, 应将该位置1。 该位置1后, 所有向接收FIFO写数据的操作将被忽略, 且系统不产生中断。 如果该位置1且SPITMDE = 0, 对接收FIFO执行读操作可以启动一次数据传输。 该位清0时, 禁用接收FIFO清空。
11	SPICONT	连续传输使能。 用户将该位置1, 连续发送使能。在主机模式下, 数据传输连续进行, 直到SPITX寄存器内无有效数据为止。 \overline{CS} 置位, 并在每一次8位串行传输期间保持置位, 直到SPITX为空。 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。

位	名称	描述
9	SPIOEN	从机MISO输出使能位。 将该位置1, 以便让MISO在正常模式下工作。 将该位清0, 可禁用MISO引脚上的输出驱动。该位被清0后, MISO引脚变为开漏极。
8	SPIROW	SPIRX上溢覆盖使能。 用户将该位置1, 新接收到的串行数据覆盖SPIRX寄存器中的有效数据。 用户清0, 新接收到的串行数据会被丢弃。
7	SPIZEN	发送FIFO为空时, SPI发送0。 若将该位置1, 则在发送FIFO无有效数据时, SPI发送0x00。 若将该位清0, 则在发送FIFO无有效数据时, SPI发送上次发送的值。
6	SPLITMDE	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SPITX为空时产生中断。 用户清0, 读取SPIRX寄存器开始发送。只有当SPIRX已满时产生中断。
5	SPIILF	LSB优先传输使能位。 用户将该位置1, 先发送LSB。 用户清0, 先发送MSB。
4	SPIWOM	SPI线或模式使能位。 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	SPICPO	串行时钟极性模式位。 用户将该位置1, 串行时钟高电平空闲。 用户清0, 串行时钟低电平空闲。
2	SPICPH	串行时钟相位模式位。 用户将该位置1, 串行时钟脉冲出现在每一次串行位传输的起始位置。 用户清0, 串行时钟脉冲出现在每一个串行位传输的末尾。
1	SPIMEN	主机模式使能位。 用户将该位置1, 主模式使能。 用户清0, 从模式使能。
0	SPIEN	SPI使能位。 用户将该位置1, SPI使能。 用户清0, SPI禁用。

ADuC7124/ADuC7126

I²C

ADuC7124/ADuC7126集成两个I²C外设，用户可将这些设备配置成完全I²C兼容型I²C总线主机或者完全I²C总线兼容型从机。两个I²C通道完全相同因此，下面的说明对两个通道均适用。

引脚SDA和SCL用于数据传输，通过对这两个引脚进行“线与”配置，可以在多主机系统中进行仲裁。这两个引脚需要接外部上拉电阻。上拉电阻的典型阻值范围为4.7 kΩ至10 kΩ。

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)；如果主机没有仲裁失效且从机进行了应答，那么开始向从机传输数据；传输会持续到主机发送一个停止条件为止，然后总线进入空闲状态。

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

ADuC7124/ADuC7126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ADuC7124/ADuC7126进行编程，可以允许其重复启动一次。在从机模式下，ADuC7124/ADuC7126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，ADuC7124/ADuC7126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。
- 在主机模式和从机模式下，均支持时钟延展。
- 在从机模式下，通过对ADuC7124/ADuC7126进行编程，可以允许其返回一个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。
- 在回送模式下，支持内、外部回送以用于I²C硬试。
- 在主机和从机模式下，发送和接收电路均有两字节的FIFO缓存。为用户提供状态位，以便控制上述FIFO缓存。

I²C功能的外部引脚配置

ADuC7124/ADuC7126有四个I²C引脚，P1.0和P1.1用于I2C0，P1.2和P1.3用于I2C1。

P1.0和P1.2的功能是传输I²C时钟信号，而P1.1和P1.3的功能是传输I²C数据信号。例如，要配置I2C0引脚(SCL0和SDA0)，GP1CON寄存器的位0和位4必须置为1，以启用I²C模式。而要配置I2C1引脚(SCL1和SDA1)，GP1CON寄存器的位8和位12必须置为1，以启用I²C模式，如“通用输入/输出”部分所示。

串行时钟发生器

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

应该配置DIVH = DIVL = 0xCF

如果希望串行时钟为400 kHz，

DIVH = 0x28, 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(传输方向位)被忽略。

ADuC7124/ADuC7126还支持10位寻址模式。当I2CxSCON寄存器的位1(ADR10EN)置1时，在从机模式下，系统支持10位地址，且将该地址保存在I2CxID0寄存器和I2CxID1寄存器之中。10位地址的组成如下：

I2CxID0[0]：读/写位，不属于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。

主机模式

在主机模式下，I2CxADR0寄存器编程为存储器件的I²C地址。

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

在10位地址模式下，10位地址创建如下：

I2CxADR0[7:3]必须设置为11110b。

I2CxADR0[2:1] = 地址位[9:8]。

I2CxADR1[7:0] = 地址位[7:0]。

I2CxADR0[0]是读/写位。

I²C寄存器

I²C外设接口由数个寄存器组成，这些寄存器将在“I²C主机寄存器”部分中讨论。

I²C主机寄存器**I²C主机控制寄存器**

名称： I2C0MCON, I2C1MCON

地址： 0xFFFFF0800, 0xFFFFF0900

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102. I2CxMCON寄存器位功能描述

位	名称	描述
[15:9]		保留。这些位为保留位，不允许向其中写入数据。
8	I2CMCENI	I ² C发送完成中断使能位。 将该位置1，则当从I ² C总线上检测到停止条件时，使能中断。 将该位清0，则清除中断源。
7	I2CNACKENI	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，则清除中断源。
6	I2CALENI	I ² C仲裁失效中断使能位。 将该位置1，则当I ² C主机无法获得对I ² C总线的控制权时，使能中断。 将该位清0，则清除中断源。
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，则清除中断源。
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	I2CMSEN	I ² C主机SCL伸展使能位。 将该位置1，可使能时钟延展功能。当SCL处于低电平状态，将该位置1，可强制器件在I2CMSEN被清0之前让SCL保持低电平状态。当SCL处于高电平状态，通过将该位置1，可强制器件在下一个下降沿后让SCL保持低电平状态。 该位清0，可禁用时钟延展功能。
2	I2CILEN	I ² C内部回送使能位。 该位将该位置1，可以使能回送测试模式。在该模式下，SCL和SDA信号在内部分别与\各自的输入信号相连。 用户清0，禁用回送模式。
1	I2CBD	I 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)。 此位清0，则等待直到I ² C总线释放。
0	I2CMEN	I ² C主机使能位。 用户将该位置1，使能I ² C主机模式。 此位清0，禁用I ² C主机模式。

ADuC7124/ADuC7126

I²C主机状态寄存器

名称:	I2C0MSTA, I2C1MSTA
地址:	0xFFFF0804, 0xFFFF0904
默认值:	0x0000, 0x0000
访问类型:	只读
功能:	该16位寄存器是主机模式下的I ² C状态寄存器。

表103. I2CxMSTA MMR位功能描述

位	名称	描述
[15:11]		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I2CxMCON寄存器的I2CMCENI位被将该位置1, 当该位的值为1时, 使能中断。 将该位清0, 则清除中断源。
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I2CxMCON寄存器的I2CNACKENI位被将该位置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主机无法获得对I ² C总线的控制权时, 此位置1。 如果I2CxMCON寄存器的I2CALENI位被将该位置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。 当主机接收到一个与地址对应的NACK条件后, 该位被置1。 如果I2CxMCON寄存器的I2CNACKENI位被将该位置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I2CxMCON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I2CxMCON寄存器的I2CMTENI位被将该位置1, 当该位被置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。

I²C主机接收寄存器

名称: I2C0MRX, I2C1MRX
 地址: 0xFFFF0808, 0xFFFF0908
 默认值: 0x00
 访问类型: 只读
 功能: 该8位寄存器是I²C主机接收寄存器。

I²C主机发送寄存器

名称: I2C0MTX, I2C1MTX
 地址: 0xFFFF080C, 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: 0xFFFF0810, 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。

表104. I2CxMCNT0寄存器位功能描述

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

I²C主机当前读取计数寄存器

名称: I2C0MCNT1, I2C1MCNT1
 地址: 0xFFFF0814, 0xFFFF0914
 默认值: 0x00, 0x00
 访问类型: 只读
 功能: 该8位寄存器用于保存在一次读序列中主机从从机处接收到的字节数。

I²C地址0寄存器

名称: I2C0ADR0, I2C1ADR0
 地址: 0xFFFF0818, 0xFFFF0918
 默认值: 0x00
 访问类型: 读/写
 功能: 当主机开始与从机进行通信后, 该8位寄存器用于保存7位从机地址和读/写位。

表105. 7位地址模式下的I2CxADR0寄存器

位	名称	描述
[7:1]	I2CADR	这些位包含目标从机的7位地址。
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106. 10位地址模式下的I2CxADR0寄存器

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

ADuC7124/ADuC7126

I²C地址1寄存器

名称: I2C0ADR1, I2C1ADR1
 地址: 0xFFFF081C, 0xFFFF091C
 默认值: 0x00
 访问类型: 读/写
 功能: 该8位寄存器只可用于10位寻址模式。
 它包含地址的最低有效字节。

表107. 10位地址模式下的I2CxADR1寄存器

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

I²C主机时钟控制寄存器

名称: I2C0DIV, I2C1DIV
 地址: 0xFFFF0824, 0xFFFF0924
 默认值: 0x1F1F
 访问类型: 读/写
 功能: 该寄存器用于控制SCL引脚上由主机产生的I²C时钟的频率。详情见I²C部分。

表109. I2CxSCON寄存器位功能描述

位	名称	描述
[15:11]		保留。
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	I2CSSEN	I ² C从机SCL延展使能位。 将该位置1, 可使能时钟延展功能。当SCL处于低电平状态, 通过将该位置1, 可强制器件在I2CMSEN被清0之前让SCL保持低电平状态。当SCL处于高电平状态, 通过将该位置1, 可强制器件在下一个下降沿后让SCL保持低电平状态。 该位清0, 可禁用时钟延展功能。

表108. I2CxDIV寄存器

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

I²C从机寄存器

I²C从机控制寄存器

名称: I2C0SCON, I2C1SCON
 地址: 0xFFFF0828, 0xFFFF0928
 默认值: 0x0000
 访问类型: 读/写
 功能: 该16位寄存器用于在从机模式下配置I²C外设。

位	名称	描述
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SCL正向沿之后立刻产生发送请求中断。 将该位置1, 则允许在读位发送的过程中, 在SCL负向沿之后立刻产生发送请求中断。
4	I2CGCCLR	I ² C广播状态与ID清除位。 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。 当该位和位2置1时, 如果已接收到一个广播信号(地址为0x00)和1字节数据, 器件将对接收寄存器中的数据和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, 可使用该功能。该广播消息会发送到总线上的所有主机。The ADuC7124/ ADuC7126会监视这些地址。要求主机注意的器件将自己的地址嵌入到消息中。所有的主机都会侦听这些消息, 然后能够处理该器件要求的主机会与其从机通信并进行相应操作。 根据2000年1月的I 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将该位置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。
1	ADR10EN	I ² C 10位地址模式。 将该位置1, 以启用10位地址模式。 将该位清0, 以启用正常地址模式。
0	I2CSEN	I ² C从机使能位。 用户将该位置1, 使能I2C从机模式。 用户将该位清0, 禁用I2C从机模式。

I²C从机状态寄存器

名称: I2C0SSTA, I2C1SSTA
地址: 0xFFFFF082C, 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110. I2CxSSTA寄存器位功能描述

位	名称	描述
15		保留。
14	I2CSTA	当检测到匹配地址后面的起始条件并接收了起始字节(0x01)后, 或者当产生了一个广播且接收到广播的代码0x00后, 该位被置1。 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。

ADuC7124/ADuC7126

位	名称	描述
[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I2CxSCON的I2CSSEN1位将该位置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I2CxSCON的I2CGCLR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I2CxSCON的I2CGCLR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。从机因发送FIFO内无数据而返回一非应答信号，或者I2CxSCON寄存器的I2CNACKEN位被将该位置1，则此位置位。 在其它条件下，该位被清0。
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。 当I2CxSCON的I2CSRXEN1位被置1时，该I2CSRXQ位可产生中断。 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I2CxSCON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I2CxSCON寄存器的I2CSETEN位将该位置1，则在读位发送的过程中，SCL正向沿之后，此位立即被置1。 I2CxSCON寄存器的I2CSTXEN1位置1时，此位可产生中断。 在其它条件下，该位被清0。
1	I2CSTFE	I ² C从机FIFO下溢状态位。 主机要求从机发送数据时，如果发送FIFO为空，则该位变为高电平。在读位操作期间，在SCL上升沿对该位进行断言。 在其它条件下，该位被清0。
0	I2CETSTA	I ² C从机传输前FIFO状态位。 如果I2CxSCON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I2CxSCON寄存器的I2CSETEN位将该位置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，该位只能被断言一次。 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: 0xFFFF0830, 0xFFFF0930
 默认值: 0x00
 访问类型: 读取
 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: 0xFFFF0834, 0xFFFF0934
 默认值: 0x00
 访问类型: 写入
 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: 0xFFFF0838, 0xFFFF0938
 默认值: 0x00
 访问类型: 读/写
 功能: I2CxSCON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它的作用了。

I²C从机ID寄存器

名称: I2C0IDx, I2C1IDx
 地址: 0xFFFF093C = I2C1ID0
 0xFFFF083C = I2C0ID0
 0xFFFF0940 = I2C1ID1
 0xFFFF0840 = I2C0ID1
 0xFFFF0944 = I2C1ID2
 0xFFFF0844 = I2C0ID2
 0xFFFF0948 = I2C1ID3
 0xFFFF0848 = 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。

I²C公共寄存器**I²C FIFO状态寄存器**

名称: I2C0FSTA, I2C1FSTA
 地址: 0xFFFF084C, 0xFFFF094C
 默认值: 0x0000
 访问类型: 读取
 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111. I2CxFSTA寄存器位功能描述

位	名称	描述
[15:10]		保留。
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]	I2CSRXSTA	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。

ADuC7124/ADuC7126

PWM概述

ADuC7124/ADuC7126集成一个6通道PWM接口(PWM0至PWM5)。PWM输出即可用于驱动H桥或也可作为标准PWM输出。上电后，PWM输出的默认为H桥。这可以确保在默认状态下电机是关闭的。在标准PWM模式下，从三对PWM引脚内输出信号。用户既可以控制每一对输出引脚的工作周期，又可以单独控制每一个输出端的占空比。

表112. PWM寄存器

名称	功能
PWMCON0	PWM控制。
PWM0COM0	用于比较PWM输出0和PWM输出1的比较寄存器0。
PWM0COM1	用于比较PWM输出0和PWM输出1的比较寄存器1。
PWM0COM2	用于比较PWM输出0和PWM输出1的比较寄存器2。
PWM0LEN	用于PWM输出0和PWM输出1的频率控制寄存器。
PWM1COM0	用于比较PWM输出2和PWM输出3的比较寄存器0。
PWM1COM1	用于比较PWM输出2和PWM输出3的比较寄存器1。
PWM1COM2	用于比较PWM输出2和PWM输出3的比较寄存器2。
PWM1LEN	用于PWM输出2和PWM输出3的频率控制寄存器。
PWM2COM0	用于比较PWM输出4和PWM输出5的比较寄存器0。
PWM2COM1	用于比较PWM输出4和PWM输出5的比较寄存器1。
PWM2COM2	用于比较PWM输出4和PWM输出5的比较寄存器2。
PWM2LEN	用于PWM输出4和PWM输出5的频率控制寄存器。
PWMCON1	PWM控制寄存器。
PWMCLR1	PWM中断清除寄存器。

PWMxCOMx寄存器用于控制在各种模式下改变PWM输出状态的时间点。第一对PWM输出示例(PWM0和PWM1)如图49所示。

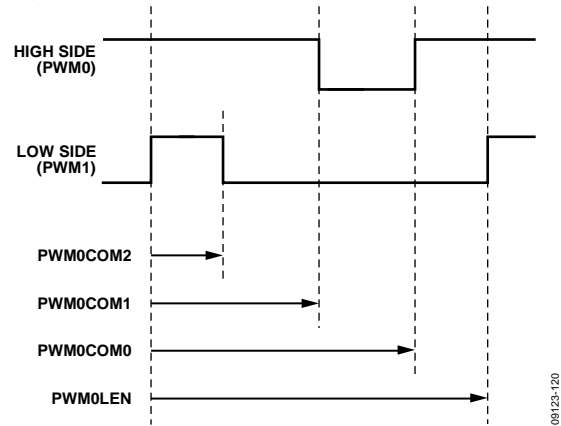


图49. PWM时序

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

PWM波形由16位定时器的计数值和比较寄存器的内容来决定，如图49所示的PWM0和PWM1的波形。

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

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

表113. PWMCON0寄存器位功能描述

位	名称	描述
14	SYNC	使能PWM同步功能。 如果用户将该位置1, 当检测到P3.7/PWM _{SYNC} 引脚上的由高到低跃迁后, 所有PWM计数器将在下一个时钟沿复位。 如果用户将该位清0, 则忽略P3.7/PWM _{SYNC} 引脚上发生的跃迁。
13	PWM5INV	通过将该位置1, 用户可将PWM5反相。 通过将该位清0, 用户可将PWM5置于正常模式下。
12	PWM3INV	通过将该位置1, 用户可将PWM3反相。 通过将该位清0, 用户可将PWM3置于正常模式下。
11	PWM1INV	通过将该位置1, 用户可将PWM1反相。 通过将该位清0, 用户可将PWM1置于正常模式下。
10	PWMTRIP	通过将该位置1, 用户可启用PWM触发中断。当PWM触发输入(引脚P3.6/PWM _{TRIP} 或引脚P0.4/PWM _{TRIP})处于低电平时, PWMEN位被清0, 且产生中断。 通过将该位清0, 用户可禁用PWM _{TRIP} 中断。
9	ENA	当HOFF = 0且HMODE = 1时, 该位可用。请注意, 该位仅在H桥模式下有效。 通过将该位置1, 用户可启用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114。
[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, 用户可反相所有PWM输出。 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PWM0和PWM2输出高电平信号。这样做还可强制PWM1和PWM3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后, 当PWM2和PWM3处于低电平状态时, 使能PWM0和PWM1作为输出信号。 用户清0后, 当PWM0和PWM1处于低电平状态时, 使能PWM2和PWM3作为输出信号。
1	HMODE	启用H桥模式 ¹ 。 通过将该位置1, 用户可启用H桥模式和PWMCON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, 用户可启用全部PWM输出。 通过将该位清0, 用户可禁用全部PWM输出。

¹ 在H桥模式下, HMODE = 1。关于PWM输出选择的说明, 参见表114。

ADuC7124/ADuC7126

表114. PWM输出选择, HMODE = 1

PWMCON0 MMR ¹				PWM输出 ²			
ENA	HOFF	POINV	DIR	PWM0	PWM1	PWM2	PWM3
0	0	X	X	1	1	1	1
X	1	X	X	1	0	1	0
1	0	0	0	0	0	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

¹X = 无关位。

²HS = 高端; LS = 低端。

上电后, PWMCON0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115)。

表115. 比较寄存器

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

通过向PWMCLRI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PWMCON1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116. PWMCON1寄存器位功能描述
(地址 = 0xFFFF0FB4; 默认值 = 0x00)**

位	值	名称	描述
7		CSEN	用户置1, 使能PWM产生转换开始信号。 用户清0, 禁用PWM转换开始信号。
[3:0]		CSD3 CSD2 CSD1 CSD0	转换开始延迟。将转换开始信号延迟数个时钟脉冲。
	0000		4个时钟脉冲。
	0001		8个时钟脉冲。
	0010		12个时钟脉冲。
	0011		16个时钟脉冲。
	0100		20个时钟脉冲。
	0101		24个时钟脉冲。
	0110		28个时钟脉冲。
	0111		32个时钟脉冲。
	1000		36个时钟脉冲。
	1001		40个时钟脉冲。
	1010		44个时钟脉冲。
	1011		48个时钟脉冲。
	1100		52个时钟脉冲。
	1101		56个时钟脉冲。
	1110		60个时钟脉冲。
	1111		64个时钟脉冲。

计算从转换开始延迟到ADC转换开始的时间时, 用户必须考虑内部延迟。下例是延迟4个时钟的情况。需要一个额外时钟来将转换开始信号传送到ADC逻辑。一旦ADC逻辑收到转换开始信号, ADC就会在下一ADC时钟沿开始转换(见图50)。

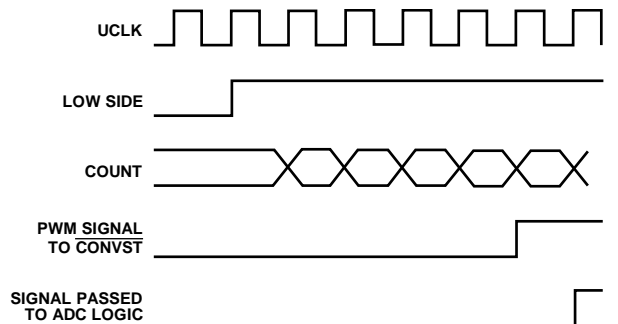


图50. ADC转换

可编程逻辑阵列(PLA)

每个ADuC7124/ADuC7126都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询表，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51所示。

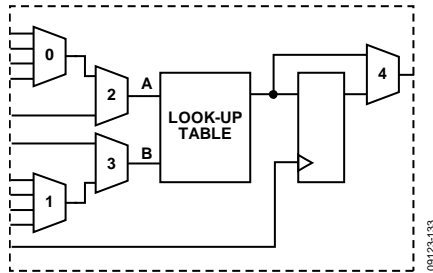


图51. PLA单元

ADuC7124/ADuC7126上共有40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 $\overline{\text{CONV}}_{\text{START}}$ 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117. 单元输入/输出¹

PLA模块0			PLA模块1		
单元	输入	输出	单元	输入	输出
0	P1.0	P1.7	8	P3.0	P4.0
1	P1.1	P0.4	9	P3.1	P4.1
2	P1.2	P0.5	10	P3.2	P4.2
3	P1.3	P0.6	11	P3.3	P4.3
4	P1.4	P0.7	12	P3.4	P4.4
5	P1.5	P2.0	13	P3.5	P4.5
6	P1.6	P2.1	14	P3.6	P4.6
7	P0.0	P2.2	15	P3.7	P4.7

¹ 此表中的所有引脚并非都连接到外部引脚，它们可能通过PLA内部路由。详情参见表122。

PLA寄存器接口

PLA外设接口由22个寄存器组成。

表118. 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。通过这些寄存器，可以配置每个单元的输入和输出多路复用器、在查询表中选择功能和选择旁路或使用触发器(参见表119和表122)。

表119. PLAELMx寄存器位功能描述

位	值	描述
[31:11]		保留。
[10:9]		Mux0控制位(见表122)。
[8:7]		Mux1控制位(见表122)。
6		Mux2控制位。 置1，选择Mux 0的输出。清0，从PLADIN中选择位值。
5		Mux3控制位。 置1，选择特定单元的输入引脚。 清0，选择Mux1的输出。
[4:1]		查询表控制位。
	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)。

ADuC7124/ADuC7126

PLACLK寄存器

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

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120. PLACLK寄存器位功能描述

位	值	描述
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	OCLK (32.768 kHz).
	101	定时器1溢出。
	110	UCLK.
	111	内部32.768振荡器。
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	OCLK (32.768 kHz).
	101	定时器1溢出。
	Other	保留。

PLAIRQ寄存器

名称: PLAIRQ
地址: 0xFFFF0B44
默认值: 0x00000000
访问类型: 读/写

PLAIRQ可以使能IRQ0和/或IRQ1和选择IRQ中断源。

表121. PLAIRQ寄存器位功能描述

位	值	描述
[15:13]		保留。
12		PLA IRQ1使能位。 置1，使能PLA输出IRQ1。 清0，禁用PLA输出IRQ1。
[11:8]		PLA IRQ1中断源。
	0000	PLA单元0。
	0001	PLA单元1。
	1111	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]		PLA IRQ0中断源。
	0000	PLA单元0。
	0001	PLA单元1。
	1111	PLA单元15。

表122. 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

PLAADC寄存器

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

PLAADC是配置PLA作为ADC起始转换信号来源的寄存器。

表123. PLAADC寄存器位功能描述

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

PLADIN寄存器

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

PLADIN是一个PLA的数据输入寄存器。

表124. PLADIN寄存器位功能描述

位	描述
[31:16]	保留。
[15:0]	单元15至单元0的输入位。

PLADOUT寄存器

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

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125. PLADOUT寄存器位功能描述

位	描述
[31:16]	保留。
[15:0]	单元15至单元0的输出位。

PLALCK寄存器

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

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

处理器相关外设

中断系统

ADuC7124/ADuC7126拥有由中断控制器控制的25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内核可以识别两类中断：正常中断请求(IRQ)和快速中断请求(FIQ)。所有中断都可以被单独屏蔽。

器件内部有许多用于控制和配置中断系统的寄存器。每一个IRQ和FIQ寄存器中的控制位都代表相同的中断源，如表126所示。

ADuC7124/ADuC7126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。中断嵌套功能必须通过设置IRQCONN寄存器的ENIRQN位使能。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126. IRQ/FIQ寄存器位功能描述

位	描述	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或唤醒定时器。
6	闪存控制0。	模块0中断的闪存控制器。
7	闪存控制1。	模块1中断的闪存控制器。
8	ADC	ADC中断源位。
9	UART0	UART0中断源位。
10	UART1	UART1中断源位。
11	PLL锁定	PLL锁定位。
12	I2C0主机IRQ	I ² C主机中断源位。
13	I2C0从机IRQ	I ² C从机中断源位。
14	I2C1主机IRQ	I ² C主机中断源位。
15	I2C1从机IRQ	I ² C从机中断源位。
16	SPI	SPI中断源位。
17	XIRQ0 (GPIO IRQ0)	外部中断0。
18	比较器	电压比较器中断源位。
19	PSM	电源监控器。
20	XIRQ1 (GPIO IRQ1)	外部中断1。

位	描述	注释
21	PLA IRQ0	PLA模块0 IRQ位。
22	XIRQ2 (GPIO IRQ2)	外部中断2。
23	XIRQ3 (GPIO IRQ3)	外部中断3。
24	PLA IRQ1	PLA模块1 IRQ位。
25	PWM	PWM触发中断源位。

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。下面说明4个专门用于IRQ的32位寄存器。

IRQSTA寄存器

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

IRQSTA寄存器

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

IRQSIG寄存器

IRQSIG反映不同IRQ源的状态。如果一个外设产生了一个IRQ信号，IRQSIG中相应的位就会被置1；否则就会被清0。当特定外设的中断请求取消时，IRQSIG的位就会被清0。通过设置IRQEN寄存器，可屏蔽所有IRQ中断源。IRQSIG为只读寄存器。中断服务程序中不应使用此寄存器来确定IRQ异常的来源，只应使用IRQSTA。

IRQSIG寄存器

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

IRQEN寄存器

IRQEN提供当前使能屏蔽的值。将该寄存器的某一位置1，可启用相应的中断请求，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寄存器

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

只能在下列情况下使用此寄存器来禁用中断源：

- 在中断源中断服务程序中。
- 外设被其自己的控制寄存器暂时禁用。

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

IRQCLR寄存器

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

快速中断请求(FIQ)

快速中断请求(IRQ)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。一个中断源可以被IRQEN屏蔽和FIQEN屏蔽同时禁用。

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，可启用相应的中断请求，此时将产生FIQ异常。当某一位被清0时，相应的中断源就会被禁止或屏蔽，此时将无法产生FIQ异常。FIQEN寄存器无法用来禁用中断。

FIQEN寄存器

名称： FIQEN
地址： 0xFFFF0108
默认值： 0x00000000
访问类型： 读/写

FIQCLR

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

只能在下列情况下使用此寄存器来禁用中断源：

- 在中断源中断服务程序中。
- 外设被其自己的控制寄存器暂时禁用。

如果IRQ源具有未处理中断或可能具有未处理中断，则不应使用该寄存器来禁用该IRQ源。

ADuC7124/ADuC7126

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: 只写

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写入IRQSTA和IRQSIG寄存器和/或FIQSTA和FIQSIG寄存器。

这个32位寄存器SWICFG专门用于软件中断(见表127)。通过该存储器映射寄存器可以对编程中断源进行控制。

表127. SWICFG寄存器位功能描述

位	描述
[31:3]	保留。
2	可编程中断FIQ。通过将该位置1或清0，可将FIQSTA和FIQSIG寄存器的位1置1或清0。
1	可编程中断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。

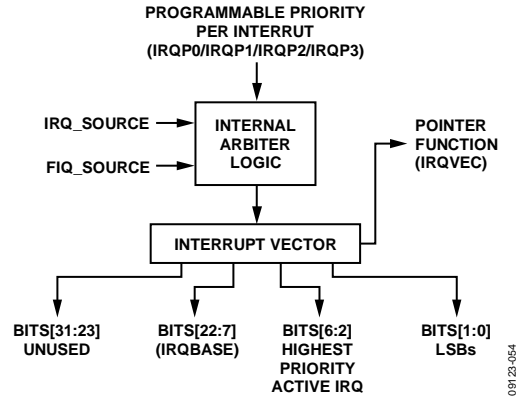


图52. 中断结构

矢量中断控制器(VIC)

ADUC7124/ADuC7126集成一个增强的中断控制系统或矢量中断控制器。通过设置IRQCONN寄存器的位0，可以启用针对IRQ中断源的矢量中断控制器。同样，通过设置IRQCONN寄存器的位1，可以启用针对FIQ中断源的矢量中断控制器。矢量中断控制器在以下几个方面增强了标准IRQ/FIQ中断：

- 矢量中断—允许用户为每个中断源单独定义中断服务程序地址。这可以通过IRQBASE和IRQVEC寄存器来完成。
- IRQ/FIQ中断—可根据优先级进行嵌套，最多允许8级嵌套。FIQ中断的优先级高于IRQ中断。因此，当使能FIQ和IRQ的矢量中断控制器且优先级为最高时，将有可能形成16个不同的中断级。
- 可编程中断优先级—通过设置IRQP0至IRQP3寄存器，可以为中断源分配不同的中断优先级(0至7)。

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: 0xFFFFF0014
 默认值: 0x00000000
 访问类型: 读/写

表128. IRQBASE寄存器位功能描述

位	类型	初始值	描述
[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129. IRQVEC寄存器位功能描述

位	类型	初始值	描述
[31:23]	R	0	读数始终为0。
[22:7]	R/W	0	IRQBASE寄存器值。
[6:2]	R	0	最高优先级中断源。这些位存储的数据对应0至27其中的一个数字, 代表可能的中断源。例如, 如果当前活动的最高优先级IRQ是定时器2, 那么这些位的值为[00100]。
[1:0]	保留	0	保留位。

优先级寄存器

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。

IRQP0寄存器

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

表130. IRQP0寄存器位功能描述

位	名称	描述
31		保留。
[30:28]	Flash1PI	为闪存模块1控制器中断源设置中断优先级(0至7)。
27		保留。
[26:24]	Flash0PI	为闪存模块0控制器中断源设置中断优先级(0至7)。
23		保留。
[22:20]	T3PI	为定时器3设置中断优先级(0至7)。
19		保留。

位	名称	描述
[18:16]	T2PI	为定时器2设置中断优先级(0至7)。
15		保留。
[14:12]	T1PI	为定时器1设置中断优先级(0至7)。
11		保留。
[10:8]	TOPI	为定时器0设置中断优先级(0至7)。
7		保留。
[6:4]	SWINTP	为软件中断源设置中断优先级(0至7)。
[3:0]		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131. IRQP1寄存器位功能描述

位	名称	描述
31		保留。
[30:28]	I2C1SPI	为I2C1从机设置中断优先级(0至7)。
27		保留。
[26:24]	I2C1MPI	为I2C1主机设置中断优先级(0至7)。
23		保留。
[22:20]	I2C0SPI	为I2C0从机设置中断优先级(0至7)。
19		保留。
[18:16]	I2C0MPI	为I2C0主机设置中断优先级(0至7)。
15		保留。
[14:12]	PLLPI	为PLL锁定中断设置中断优先级(0至7)。
11		保留。
[10:8]	UART1PI	为UART1设置中断优先级(0至7)。
7		保留。
[6:4]	UART0PI	为UART0设置中断优先级(0至7)。
5		保留。
[2:0]	ADCPI	为ADC中断源设置中断优先级(0至7)。

ADuC7124/ADuC7126

IRQP2寄存器

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

表132. IRQP2寄存器位功能描述

位	名称	描述
31		保留。
[30:28]	IRQ3PI	为IRQ3设置中断优先级(0至7)。
27		保留。
[26:24]	IRQ2PI	为IRQ2设置中断优先级(0至7)。
23		保留。
[22:20]	PLA0PI	为PLA IRQ0设置中断优先级(0至7)。
19		保留。
[18:16]	IRQ1PI	为IRQ1设置中断优先级(0至7)。
15		保留。
[14:12]	PSMPI	为电源监控器中断源设置中断优先级(0至7)。
11		保留。
[10:8]	COMPI	为比较器设置中断优先级(0至7)。
7		保留。
[6:4]	IRQ0PI	为IRQ0设置中断优先级(0至7)。
3		保留。
[2:0]	SPIPI	为SPI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133. IRQP3寄存器位功能描述

位	名称	描述
[31:7]		保留。
[6:4]	PWMPI	为PWM设置中断优先级(0至7)。
3		保留。
[2:0]	PLA1PI	为PLA IRQ1设置中断优先级(0至7)。

IRQCONN寄存器

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

将这两位清0，则FIQ和IRQ仍可使用，但无法嵌套IRQ或

FIQ，也无法设置中断源优先级。在此默认状态下，FIQ的优先级高于IRQ的优先级。

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

表134. IRQCONN寄存器位功能描述

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

IRQSTAN寄存器

如果IRQCONN位0被置位且IRQVEC被读取，则IRQSTAN[7:0]中的一位将被置位。IRQ的优先级决定了第几位将被置位。如果IRQ中断的优先级为0，则位0被置位；如果IRQ中断的优先级为1，则位1被置位；以此类推。将该寄存器中的某一位置1，可屏蔽所有与该位所对应的中断具有相同或更低优先级的中断。

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

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

表135. IRQSTAN寄存器位功能描述

位	名称	描述
31:8		保留。这些位为保留位，不允许向其中写入数据。
7:0		将这些位置1，可以使能FIQ中断嵌套。将这些位清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： 0xFFFFF011C
默认值： 0x00000000
访问类型： 只读

表136. FIQVEC寄存器位功能描述

位	类型	初始值	描述
[31:23]	R	0	读数始终为0。
[22:7]	R/W	0	IRQBASE寄存器值。
[6:2]		0	最高优先级中断源。这是一个介于0至27的值，代表当前活动的中断源。中断如表126所示。例如，如果当前活动的优先级最高的FIQ是定时器2，那么这些位的值为[00100]。
[1:0]		0	保留。

FIQSTAN寄存器

如果IRQCONN位1被置位且FIQVEC被读取，则FIQSTAN[7:0]中的一位将被置位。FIQ的优先级决定了第几位将被置位。如果FIQ的优先级为0，则位0被置位；如果FIQ中断的优先级为1，则位1被置位；以此类推。

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

如需清0该寄存器中的某一位，必须先将其它与更高优先级对应的位清0。一次只能清0一位。例如，如果该寄存器

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： 0xFFFFF013C
默认值： 0x00000000
访问类型： 读/写

表137. FIQSTAN寄存器位功能描述

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

外部中断和PLA中断

ADuC7124/ADuC7126最多可支持4个外部中断源和2个PLA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。

要使能外部中断源或PLA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。

通过设置IRQCLRE中的适当位，可以清除边沿触发的外部IRQ中断或边沿触发的PLA中断。

IRQCONE寄存器

名称： IRQCONE
地址： 0xFFFFF0034
默认值： 0x00000000
访问类型： 读/写

表138. IRQCONE寄存器位功能描述

位	值	名称	描述
[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	PLA1SRC[1:0]	PLA IRQ1在下降沿触发。 PLA IRQ1在上升沿触发。 PLA IRQ1在低电平时触发。 PLA IRQ1在高电平时触发。
[9:8]	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。

ADuC7124/ADuC7126

位	值	名称	描述
[7:6]	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。
[5:4]	11 10 01 00	PLA0SRC[1:0]	PLA IRQ0在下降沿触发。 PLA IRQ0在上升沿触发。 PLA IRQ0在低电平时触发。 PLA IRQ0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139. IRQCLRE寄存器位功能描述

位	名称	描述
[31:25]		保留。这些位为保留位，不允许向其中写入数据。
24	PLA1CLRI	必须在PLA IRQ1中断服务程序中在该位置1，以清除边沿触发的PLA IRQ1中断。
23	IRQ3CLRI	必须在外部IRQ3中断服务程序中在该位置1，以清除边沿触发的IRQ3中断。
22	IRQ2CLRI	必须在外部IRQ2中断服务程序中在该位置1，以清除边沿触发的IRQ2中断。
21	PLA0CLRI	必须在PLA IRQ0中断服务程序中在该位置1，以清除边沿触发的PLA IRQ0中断。
20	IRQ1CLRI	必须在外部IRQ1中断服务程序中在该位置1，以清除边沿触发的IRQ1中断。
[19:18]		保留。这些位为保留位，不允许向其中写入数据。
17	IRQ0CLRI	必须在外部IRQ0中断服务程序中在该位置1，以清除边沿触发的IRQ0中断。
[16:0]		保留。这些位为保留位，不允许向其中写入数据。

定时器

ADuC7124/ADuC7126具有4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

这4个定时器可以工作在两种模式：自由模式和周期模式。

在自由模式下，计数器将从最大值递减计数一直到零，并且在达到最小值后重新递增计数。它也可以从最小值开始递增计数直到满量程并且在达到最大值后重新开始递减计数。

在周期模式下，计数器以装载寄存器(TxLD寄存器)中的值为起始值，开始递减/递增计数至零或满量程，然后再以该寄存器中的值为起始值，重新开始计数。

定时器间隔可通过下式计算：

如果将定时器设置为递减计数，那么

$$Interval = \frac{(TxLD) \times Prescaler}{Source\ Clock}$$

如果将定时器设置为递增计数，那么

$$Interval = \frac{(FullScale - TxLD) \times Prescaler}{Source\ Clock}$$

通过访问计数器的值寄存器(TxVAL)，可以随时读出计数器的值。但要注意：当定时器的时钟源是内核时钟以外的时钟时，可能由于时钟系统不同步而读出错误的值。在这种情况下，TxVAL始终应该读两次。如果两次读到的结果不同，应该再读一次以得到正确的值。

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，当递减计数时，每一次计数器的值达到0时就会产生一个IRQ中断。当递增计数时，每一次计数器的值达到满量程值时也会产生一个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。

当使用异步的时钟定时器时，取消定时器模块内的中断所占用的时间要比执行中断程序中的代码所占用的时间长。在退出中断服务程序之前，要保证中断信号已经被取消。这可以通过检查IRQSTA寄存器来实现。

时:分:秒:1/128格式

定时器1和定时器2具有“时:分:秒:百位数”格式。若要使用“时:分:秒:百位数”格式，应选择32768 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(256/32768)。代表时、分、秒的位在寄存器中不是连续的。使用TxCON[5:4]设置的“时:分:秒:百位数”格式时，这一配置适用于TxLD和TxVAL。详情参见表140。

表140. 时:分:秒:百位数格式

位	名称	描述
[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

ADuC7124/ADuC7126

定时器0(RTOS定时器)

定时器0是一个带有可编程预分频器的16位通用定时器(递减计数)。预分频器源采用内核时钟频率(HCLK)，分频系数共有1、16和256三种。

定时器0可用于启动ADC转换，如框图53所示。

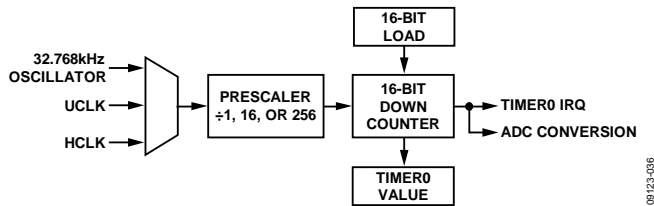


表53. 定时器0框图

定时器0接口包括4个寄存器：T0LD、T0VAL、T0CON和T0CLRI。

T0LD寄存器

名称： T0LD
地址： 0xFFFF0300
默认值： 0x0000
访问类型： 读/写

T0LD是一个16位装载寄存器。

T0VAL寄存器

名称： T0VAL
地址： 0xFFFF0304
默认值： 0xFFFF
访问类型： 只读

T0VAL是一个16位只读寄存器，用于表示计数器的当前状态。

T0CON寄存器

名称： T0CON
地址： 0xFFFF0308
默认值： 0x0000
访问类型： 读/写

T0CON是一个配置寄存器，如表141所示。

表141. T0CON寄存器位功能描述

位	值	描述
[31:8]		保留。
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。 默认模式。
[5:4]		时钟选择位。
	00	HCLK.
	01	UCLK.
	10	32.768 kHz.
	11	保留。
[3:2]		预分频选择位。
	00	内核时钟/1。默认值。
	01	内核时钟/16。
	10	内核时钟/256。
	11	未定义。与00时相同。
[1:0]		保留。

T0CLR1寄存器

名称: T0CLR1
地址: 0xFFFFF030C
默认值: 0xFF
访问类型: 只写

T0CLR1是一个8位寄存器。向其中写入任意值就可以清除中断。

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, 工作时可递增计数或递减计数。时钟源可以是32 kHz外部晶振、未分频系统、内核时钟或P1.1引脚(最高频率41.78 MHz)。时钟源分频系数共有1、16、256或32,768四种。

可以1, 1/16, 1/256或1/32768对该源时钟分频。计数器可以是标准的32位数模式或下面的形式:时:分:秒:百分之一秒。

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。这一特性可用于判断一个事件的置位, 当用于IRQ中断请求服务时, 这种方法比RTOS定时器所允许的精度更高。

定时器1可用于启动ADC转换。

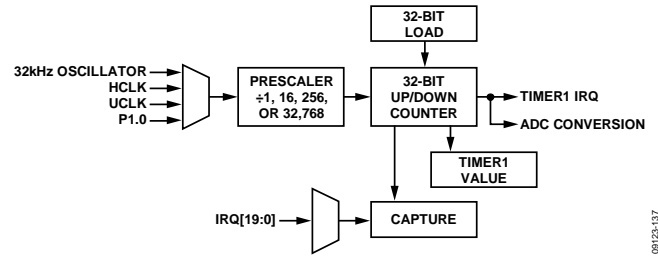


图54. 定时器1框图

定时器1接口有5个寄存器: T1LD、T1VAL、T1CON、T1CLR1和T1CAP。

T1LD寄存器

名称: T1LD
地址: 0xFFFFF0320
默认值: 0x00000000
访问类型: 读/写

T1LD是一个32位装载寄存器。

T1VAL寄存器

名称: T1VAL
地址: 0xFFFFF0324
默认值: 0xFFFFFFFF
访问类型: 只读

T1VAL是一个32位只读寄存器, 用于表示计数器的当前状态。

T1CON寄存器

名称: T1CON
地址: 0xFFFFF0328
默认值: 0x0000
访问类型: 读/写

T1CON是一个配置寄存器, 如表142所示。

ADuC7124/ADuC7126

表142. T1CON寄存器位功能描述

位	值	描述
[31:18]		保留。
17		事件选择位。 用户置1, 使能对事件的实时捕捉。 用户清0, 禁用对事件的实时捕捉。
[16:12]		事件选择范围0至25。这些事件如表126所示, 其中所有事件的序号对于定时器0都相应减少2。例如, 表126中的事件2对于定时器0来说就是事件0。
[11:9]		时钟选择。
	000	内核时钟(41 MHz/2 ^{CD})。
	001	32.768 kHz。
	010	UCLK。
	011	P1.0上升沿触发。
8		递增计数。 用户置1, 定时器1递增计数。 用户清0, 定时器1递减计数(默认)。
7		定时器1使能位。 用户置1, 使能定时器1。 用户清0, 禁用定时器1(默认)。
6		定时器1模式。 用户置1, 选择周期模式。 用户清0, 选择自由模式。默认模式。
[5:4]		格式。
	00	二进制。
	01	保留。
	10	时:分:秒:百分之一秒(23小时至0小时)。
	11	时:分:秒:百分之一秒(255小时至0小时)。
[3:0]		预分频选择位。
	0000	时钟源/1。
	0100	时钟源/16。
	1000	时钟源/256。
	1111	时钟源/32,768。

T1CLRI寄存器

名称: T1CLRI
地址: 0xFFFFF032C
默认值: 0xFF
访问类型: 只写

T1CLRI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。

T1CAP寄存器

名称: T1CAP
地址: 0xFFFFF0330
默认值: 0x00000000
访问类型: 读/写

T1CAP是一个32位寄存器。当一个特殊事件发生时, T1CAP会保持T1VAL寄存器中当时的数据。这个事件必须是在T1CON中已经选择的。

定时器2(唤醒定时器)

定时器2是一个带有可编程预分频器的32位唤醒定时器, 可递增计数或递减计数。预分频器直接从四个时钟源中的一个接收时钟信号: 内核时钟(默认选择)、内部32.768 kHz振荡器、外部32.768 kHz时钟晶体或PLL未分频时钟。所选的时钟源可经1、16、256或32768分频。当内核时钟被禁用后, 唤醒定时器仍继续工作。这样, 当内核在41.78 MHz频率下工作时, 如预分频系数为1, 定时器的最小分辨率能够达到22 ns。如果通过IRQEN[4]使能定时器2, 就可以捕捉当前定时器值(见表126)。

计数器可以采用标准的32位数格式或者时间格式(时:分:秒:百分之一秒)。

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

定时器2接口有4个寄存器, 如表143所示。

表143. 定时器2接口寄存器

寄存器	描述
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。
T2CLRI	8位寄存器。向其中写入任意值就可以清除定时器2中断。
T2CON	配置寄存器。

定时器2载入寄存器

名称: T2LD
地址: 0xFFFFF0340
默认值: 0x000000
访问类型: 读/写

T2LD是一个32位寄存器, 用于保存载入计数器的32位值。

定时器2清除寄存器

名称: T2CLRI
地址: 0xFFFFF034C
默认值: 0x00
访问类型: 只写

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。

定时器2值寄存器

名称: T2VAL
地址: 0xFFFF0344
默认值: 0x0000
访问类型: 只读

T2VAL为32位寄存器，用于保存定时器2的当前值。

定时器2控制寄存器

名称: T2CON
地址: 0xFFFF0348
默认值: 0x0000
访问类型: 读/写

该32位寄存器用于设置定时器2的工作模式。

表144. T2CON寄存器位功能描述

位	值	描述
[31:11]		保留。
10:9]		时钟源选择。
	00	外部32.768 kHz时钟晶体(默认)。
	01	外部32.768 kHz时钟晶体。
	10	内部32.768 kHz振荡器。
	11	HCLK。
8		递增计数。 用户置1，定时器2递增计数。 用户清0，定时器2递减计数(默认)。
7		定时器2使能位。 用户置1，使能定时器2。 用户清0，禁用定时器2(默认)。
6		定时器2模式。 用户置1，选择周期模式。 用户清0，选择自由模式(默认)。
[5:4]		格式。
	00	二进制(默认)。
	01	保留。
	10	时:分:秒:百分之一秒(23小时至0小时)。
	11	时:分:秒:百分之一秒(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	时钟源/256。
	1111	时钟源/32,768。

ADuC7124/ADuC7126

定时器3(看门狗定时器)

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，它需要周期服务来阻止它强迫处理器复位。

正常模式

在普通模式下，除了时钟源和递增计数功能，定时器3和定时器0的功能相同。时钟源来自于锁相环(32KHz)，其时钟分频系数共有1、16、256三种(如图55所示)。

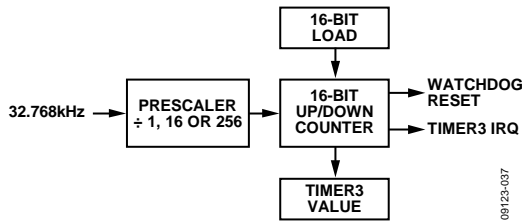


图55. 定时器3框图

看门狗模式

将T3CON MMR的第5位置1便可以进入看门狗模式。定时器3从T3LD寄存器中的值开始递减计数直到0。T3LD用作超时。当使用1/256预分频时，最大的超时时间为512秒，且为T3LD的满量程。在看门狗模式下工作时，定时器3的时钟源为内部的32KHz晶振。注意：为了成功进入看门狗模式，必须在写入T3LD MMR以后再对T3CON MMR的第5位置1。

如果定时器计数到0，就会产生一个复位或中断，这取决于T3CON寄存器的第1位的配置。如果不想产生复位或中断，必须在定时周期结束之前向T3CLR1中写入任意一个值。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。此时这两个寄存器不能被修改，直到有一个复位信号清除了看门狗使能位，这将使定时器3退出看门狗模式。

定时器3接口包括4个存储器映像寄存器：T3LD、T3VAL、T3CON和T3CLR1。

T3LD寄存器

名称： T3LD
地址： 0xFFFF0360
默认值： 0x0000
访问类型： 读/写

T3LD是一个16位装载寄存器。

T3VAL寄存器

名称： T3VAL
地址： 0xFFFF0364
默认值： 0xFFFF
访问类型： 只读

T3VAL是一个16位只读寄存器，它记录的是计数器当前的状态。

T3CON寄存器

名称： T3CON
地址： 0xFFFF0368
默认值： 0x0000
访问类型： 读/写

T3CON是配置MMR，如表145所示。

表145. T3CON MMR位功能描述

位	值	描述
[31:9]		保留。
8		递增计数。 用户置1，定时器3递增计数。 用户清0，定时器3递减计数(默认)。
7		定时器3使能位。 用户置1，使能定时器3。 用户清0，禁用定时器3(默认)。
6		定时器3工作模式。 用户置1，选择周期模式。 用户清0，选择自由模式(默认)。
5		看门狗模式使能位。 用户置1，使能看门狗模式。 用户清0，禁用看门狗模式(默认)。
4		安全清除位。 用户置1，使能安全清除。 用户清0，禁用安全清除(默认)。
[3:2]		预分频选择位。
	00	时钟源/1(默认)。
	01	时钟源/16。
	10	时钟源/256。
	11	未定义。与00时相同。
1		看门狗IRQ选项位。 用户置1，在看门狗计数值为0时产生IRQ中断而非复位。 用户清0，禁用IRQ选项。
0		保留。

T3CLRI寄存器

名称: T3CLRI
地址: 0xFFFF036C
默认值: 0x00
访问类型: 只写

T3CLRI是一个8位寄存器。在正常模式下, 向这个寄存器连续写入任意值, 将清除定时器3中断; 而在看门狗模式下向这个寄存器写入任意值, 则重新开始一个超时周期。

注意一定要确保连续写入才能开始一个新的超时周期。

安全清除位(仅用于看门狗模式)

安全清除位用于实现更高层次的保护。当它被置1时, 一个特殊的数值序列就必须写入T3CLRI中来避免看门狗复位。这个特殊的数值序列是由一个8位的线性反馈移位寄存器(LFSR)多项式 $=X_8+X_6+X_5+X+1$ 产生的, 如图56所示。

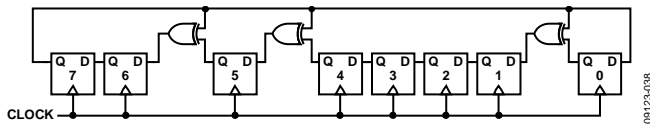


图56. 8位LFSR

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

根据这个多项式的性质, 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的值不能被浏览, 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。

外部存储器接口

ADuC7124/ADuC7126具有外部存储器接口。外部存储器接口需要大量的引脚。当使用外部端口时, XMCFG存储器映射寄存器必须被置1。

尽管内部支持32位的地址, 但外部引脚上只有低16位地址。

存储器接口可以寻址多达4个128 kB的异步存储器(SRAM或/

和EEPROM)。

一个外部存储器接口所需的引脚如表146所示。

表146. 外部存储器接口引脚

引脚	功能
AD[15:0]	地址/数据总线。
A16	仅用于8位存储器的扩展地址。
MS[3:0]	存储器选择。
\overline{WS}	写选通。
\overline{RS}	读选通。
AE	地址锁存使能。
$\overline{BHE}, \overline{BLE}$	字节写入功能。

共有4个可用的外部存储区域, 如表147所示。与每一个区域相关的引脚是MS[3:0]。这些信号允许访问外部存储器的特定区域。每一个存储区域最大为128 kB、64 kB × 16或128 kB × 8。为访问128 kB的8位存储器, 需要一条额外的地址线(A16, 参见图57示例。这4个区域可单独配置。

表147. 存储区域

起始地址	结束地址	目录
0x10000000	0x1000FFFF	外部存储区域0
0x20000000	0x2000FFFF	外部存储区域1
0x30000000	0x3000FFFF	外部存储区域2
0x40000000	0x4000FFFF	外部存储区域3

每一个外部存储区域可通过3个存储器映射寄存器控制: XMCFG、XMxCON和XMxPAR。

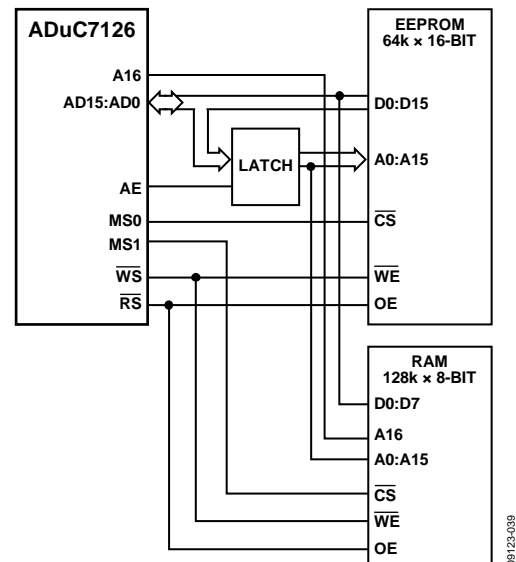


图57. 外部EEPROM/RAM接口

ADuC7124/ADuC7126

XMCFG寄存器

名称: XMCFG
地址: 0xFFFFF000
默认值: 0x00
访问类型: 读/写

将XMCFG寄存器置1可以使能访问外部存储器。要将任何端口引脚配置为外部存储器访问引脚，先必须将此寄存器置1。端口引脚也必须通过GPxCON存储器映射寄存器单独使能。

表148. XMxCON寄存器

名称	地址	默认值	访问类型
XM0CON	0xFFFFF010	0x00	R/W
XM1CON	0xFFFFF014	0x00	R/W
XM2CON	0xFFFFF018	0x00	R/W
XM3CON	0xFFFFF01C	0x00	R/W

XMxCON是每一个存储区域的控制寄存器。它们用于允许使能/禁止一个存储区域并且控制存储区域的数据总线宽度。

表149. XMxCON寄存器位功能描述

位	描述
1	选择数据总线宽度。 置1，使用16位数据总线。 清0，使用8位数据总线。
0	使能存储区域。 置1，使能存储区域。 清0，禁用存储区域。

表150. XMxPAR寄存器

名称	地址	默认值	访问类型
XM0PAR	0xFFFFF020	0x70FF	R/W
XM1PAR	0xFFFFF024	0x70FF	R/W
XM2PAR	0xFFFFF028	0x70FF	R/W
XM3PAR	0xFFFFF02C	0x70FF	R/W

XMxPAR存储器用来定义访问外部存储器的协议。

表151. XMxPAR寄存器位功能描述

位	描述
15	使能字节写选通。此位仅用于两个8位存储器模块共享同一存储区域时。 置1，使A0作为WS信号输出。这样可以在不使用BHE和BLE信号的情况下启用字节写入功能。 清0，使用BHE和BLE信号。
[14:12]	地址锁存使能选通的等待状态个数。
11	保留。
10	额外地址保持时间。 置1，禁用额外保持时间。 清0，在读和写时使能1个周期的地址保持时间。
9	读数据时额外总线转换时间。 置1，禁用额外总线转换时间。 清0，在读选通(RS)之前和之后使能1个额外时钟。
8	写数据时额外总线转换时间。 置1，禁用额外总线转换时间。 清0，在写选通(WS)之前和之后使能1个额外时钟。
[7:4]	写等待状态个数。 选择添加到WS脉冲长度的等待状态个数。0x0: 1个时钟; 0xF: 16个时钟周期(默认值)。
[3:0]	读等待状态个数。 选择添加到RS脉冲长度的等待状态个数。0x0: 1个时钟; 0xF: 16个时钟周期(默认值)。

图58、图59、图60和图61分别给出了1个读周期时序图、1个带有地址保持和总线转换周期的读周期时序图、1个带有地址和写保持周期的写周期时序图和1个带有等待状态的写周期时序图。

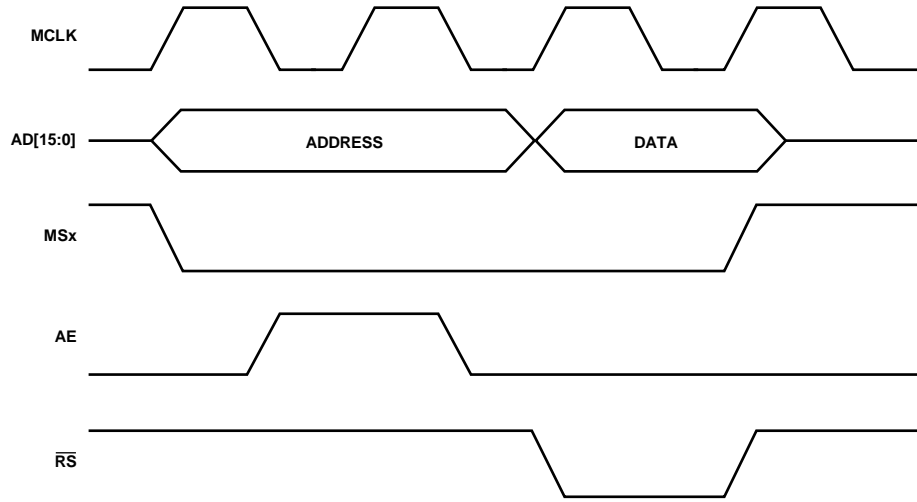


图58 外部存储器读周期

09123-040

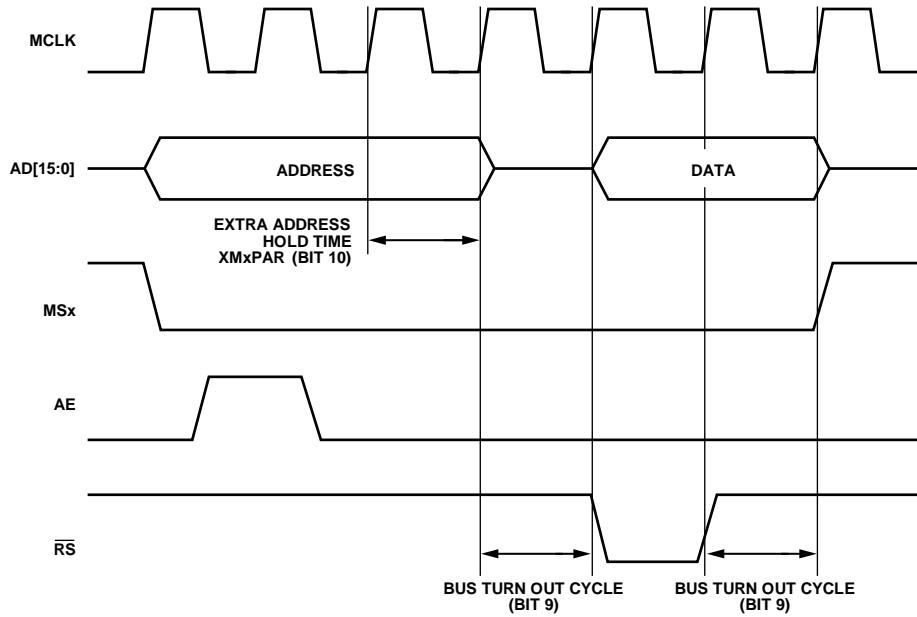


图59. 带有地址保持和总线转换周期的外部存储器读周期

09123-041

ADuC7124/ADuC7126

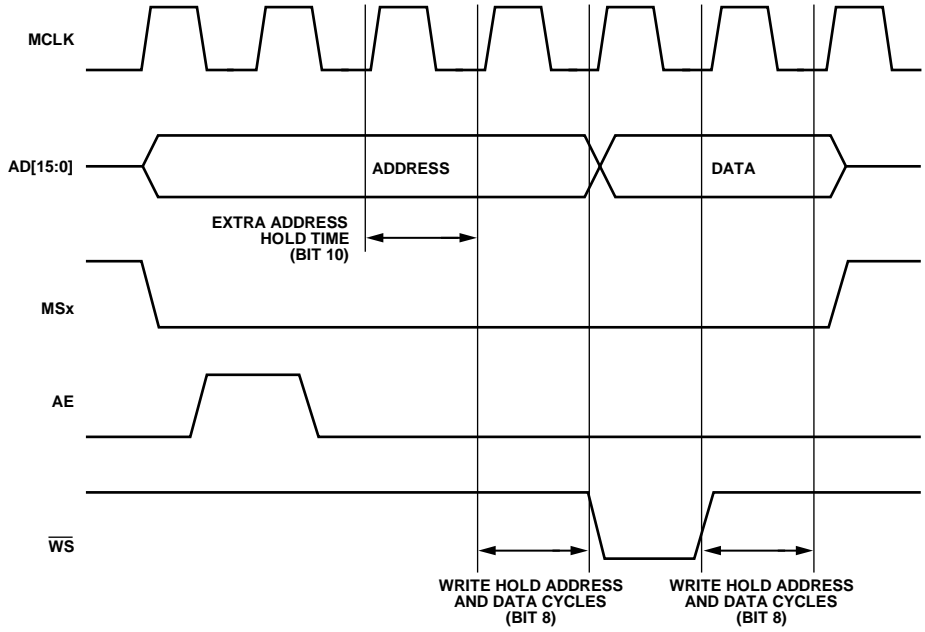


图60. 带有地址和写保持周期的外部存储器写周期

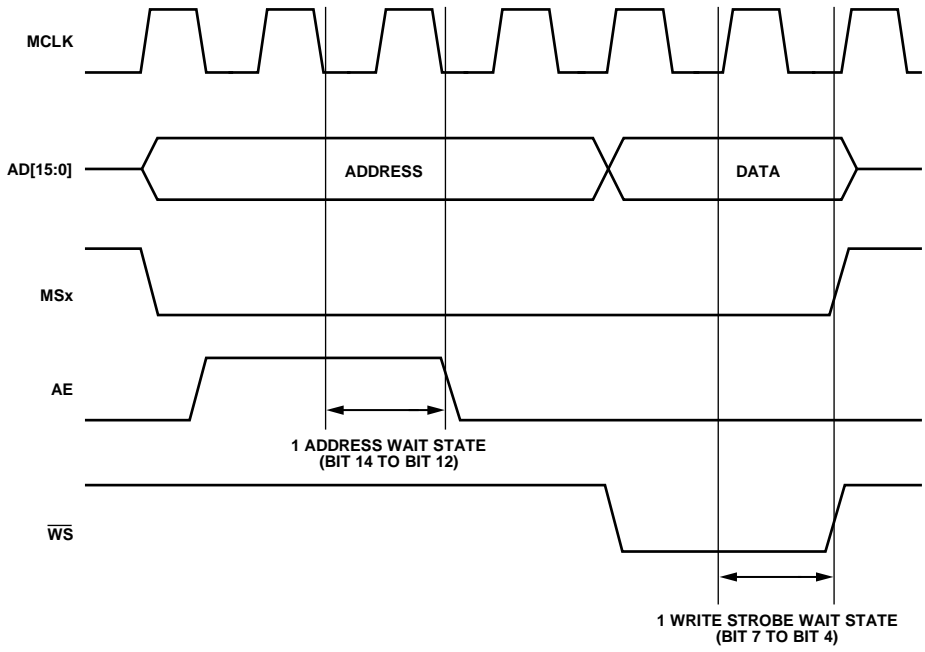


图61. 带有等待状态的外部存储器写周期

硬件设计考虑

电源

ADuC7124/ADuC7126工作电压范围为2.7 V至3.6 V。模拟电源引脚和数字电源引脚(分别对应 AV_{DD} 和 IOV_{DD})是分离的,因此 AV_{DD} 不受系统 IOV_{DD} 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,各个电源的电压可以是不同的。例如,系统的 IOV_{DD} 工作电压为3.3 V,而 AV_{DD} 电压为3 V;反之亦然。图62是一个典型的分离电源配置。

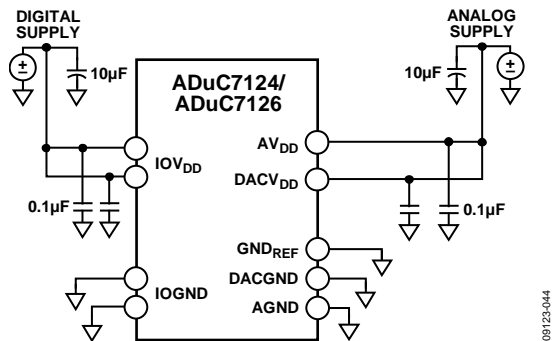


图62. 外部双电源连接

除了使用两个分离的电源外,用户还可以通过在 AV_{DD} 和 IOV_{DD} 之间串联一个小电阻和/或磁珠来降低 AV_{DD} 的噪声,然后将 AV_{DD} 单独去偶到地。图63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 AV_{DD} 供电。

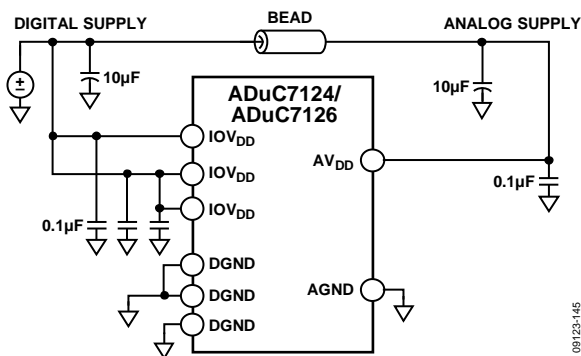


图63. 外部单电源连接

注意在图62和图63中,在 IOV_{DD} 处有一个大容量值(10 μ F)的储能电容,以及在 AV_{DD} 处单独有一个10 μ F的电容。此外,在芯片的每一个 AV_{DD} 和 IOV_{DD} 引脚都连接了一个小容值(0.1 μ F)电容。在标准设计中,必须确保包括所有这些电容且电容量越小的电容越接近各 AV_{DD} 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,ADuC7124/ADuC7126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。

IOV_{DD} 电源灵敏度

IOV_{DD} 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 IOV_{DD} 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50mV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 IOV_{DD} 上的所有噪声低于50 mV,那么就需要一个图64所推荐的滤波电路。

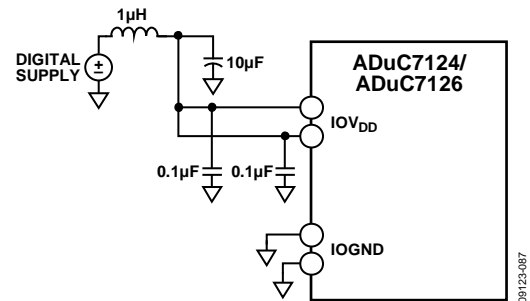


图64. 推荐的 IOV_{DD} 电源滤波电路

线性稳压器

ADuC7124/ADuC7126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 IOV_{DD} 的电源调节到2.6 V,从而为内核逻辑供电。 LV_{DD} 引脚的2.6 V电源用来给内核逻辑供电。在 LV_{DD} 和 $DGND$ 之间必须连接一个0.47 μ F的补偿电容(应尽量靠近这些引脚)作为电荷槽,如图65所示。

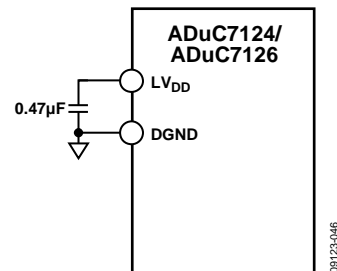


图65. 稳压器连接

LV_{DD} 引脚不能用来给任何其它芯片供电。此外,推荐在 IOV_{DD} 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。

ADuC7124/ADuC7126

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ADuC7124/ADuC7126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)，使用者一定不能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66a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(如系统电源处)，那么这个平面不能再连接到靠近器件的地方，因为这样会形成一个地环路。在这种情况下，ADuC7124/ADuC7126的所有AGND和DGND引脚都应连接到模拟地平面，如图66b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)。

ADuC7124/ADuC7126可以放置在数字和模拟部分之间，如图66c所示。

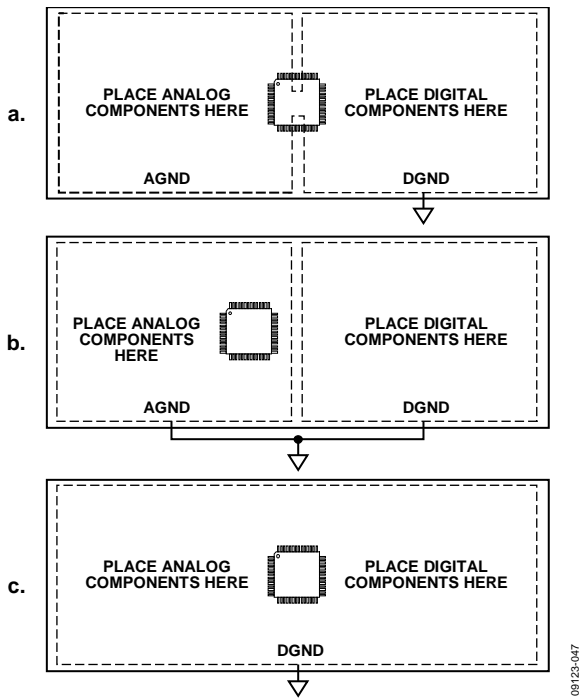


图66. 系统地框图

在上述方案以及更为复杂的实际应用中，用户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路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IOV_{DD}给模拟部分的器件供电(如图66b所示)，因为这会导致IOV_{DD}回路电流强行通过AGND。如果一个带有噪声的数字芯片被放置在板的左半平面(如图66c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ADuC7124/ADuC7126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

ADuC7124/ADuC7126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XCLKI和XCLKO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67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32.768 kHz ± 3%的典型频率。

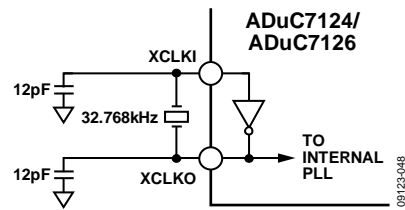


图67. 外部并行谐振晶体连接

若要使用一个外部时钟源输入来代替锁相环(如图68所示)，必须修改PLLCON寄存器的位1和位0。外部时钟从P0.7和XCLK引脚输入。

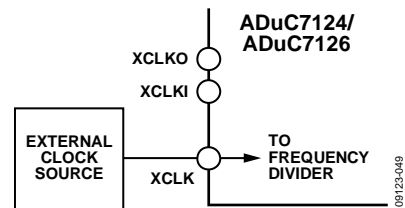


图68. 连接一个外部时钟源

使用外部时钟源时，ADuC7124/ADuC7126的额定时钟频率范围为50 kHz到41.78 MHz ± 1%，以确保模拟外设和Flash/EE正常工作。

上电复位操作

ADuC7124/ADuC7126具有一个内部上电复位(POR)电路，当 LV_{DD} 低于2.40 V典型值时，内部POR会使器件保持在复位状态。当 LV_{DD} 上升超过2.41 V时，一个内部定时器通常会在128 ms后溢出，从而使芯片脱离复位状态。用户此时必须保证给 IOV_{DD} 供电的电源电压至少稳定在2.7 V。当器件关断时，内部POR同样会使器件保持在复位状态，直到 LV_{DD} 降至2.40 V以下。图69详细说明了内部POR的工作原理。

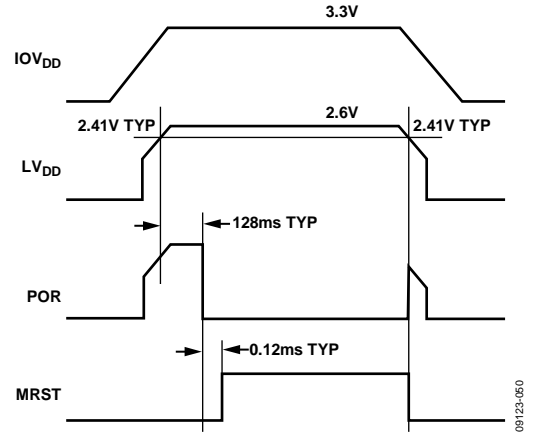
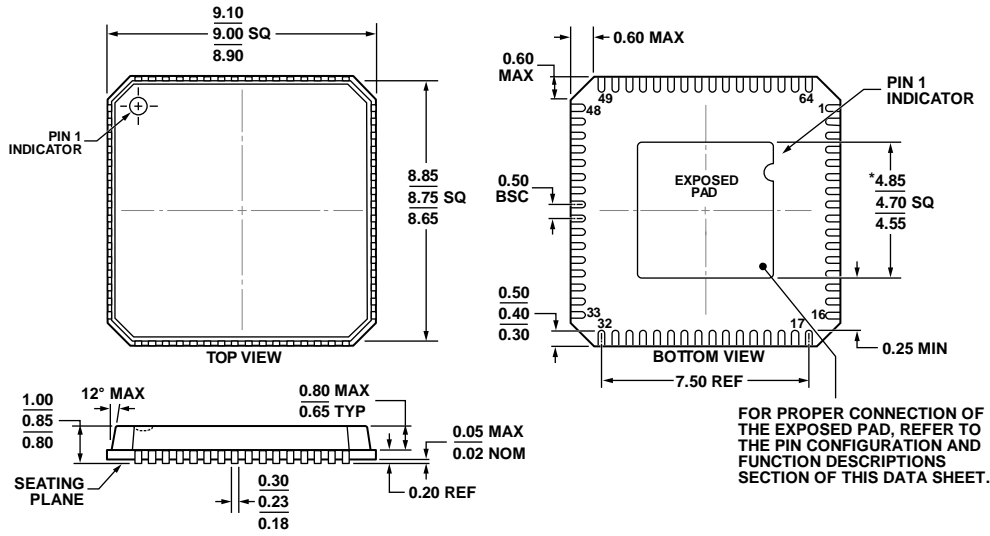


图69. 内部上电复位操作

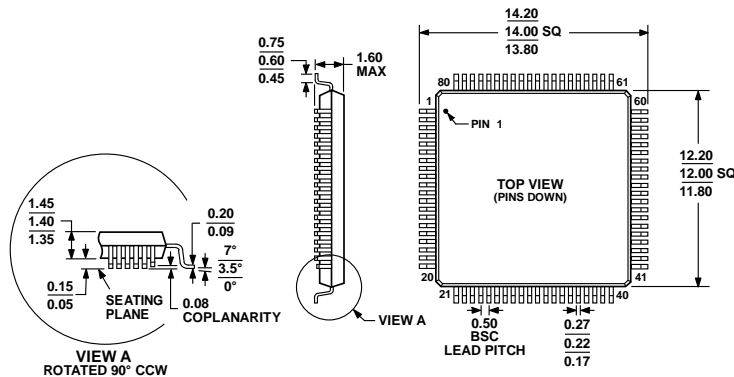
ADuC7124/ADuC7126

外形尺寸



*COMPLIANT TO JEDEC STANDARDS MO-220-VMM4-4 EXCEPT FOR EXPOSED PAD DIMENSION

图70. 64引脚引脚架构芯片级封装[LFCSP_VQ]
9 mm x 9 mm, 超薄体
(CP-64-1)
图示尺寸单位: mm



COMPLIANT TO JEDEC STANDARDS MS-026-BDD

图71. 80引脚薄型四方扁平封装[LQFP]
(ST-80-1)
尺寸单位: mm

订购指南

型号 ¹	ADC通道	DAC通道	Flash/RAM	GPIO	下载器	温度范围	封装描述	封装选项	订购数量
ADuC7124BCPZ126	10	2	126 kB/32 kB	30	UART	-40°C 至 +125°C	64引脚 LFCSP_VQ	CP-64-1	260
ADuC7124BCPZ126-RL	10	2	126 kB/32 kB	30	UART	-40°C至 +125°C	64引脚 LFCSP_VQ	CP-64-1	2500
ADuC7126BSTZ126	12	4	126 kB/32 kB	40	UART	-40°C 至 +125°C	80引脚 LQFP封装	ST-80-1	119
ADuC7126BSTZ126-RL	12	4	126 kB/32 kB	40	UART	-40°C 至 +125°C	80引脚 LQFP封装	ST-80-1	1000
ADuC7126BSTZ126I	12	4	126 kB/32 kB	40	I ² C	-40°C 至 +125°C	80引脚 LQFP封装	ST-80-1	119
ADuC7126BSTZ126IRL	12	4	126 kB/32 kB	40	I ² C	-40°C 至 +125°C	80引脚 LQFP封装	ST-80-1	1000
EVAL-ADuC7124QSPZ							ADuC7124 QuickStart 开发系统		
EVAL-ADuC7126QSPZ							ADuC7126 QuickStart 开发系统		

¹Z = 符合RoHS标准的器件。

注释

注释

注释

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