



精密模拟微控制器， 12位模拟I/O，ARM7TDMI MCU

ADuC7019/20/21/22/24/25/26/27/28/29

产品特性

模拟I/O

多通道、12位、1 MSPS ADC

最多16个ADC通道¹

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

最多4路DAC输出可用¹

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44 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

62 kB Flash/EE存储器，8 kB SRAM

在线下载，基于JTAG调试

软件触发在线重新编程能力

片内外设

UART、2个I²C[®]和SPI串行I/O端口

最多40引脚GPIO端口¹

4个通用定时器

唤醒和看门狗定时器(WDT)

电源监控器

三相、16位PWM发生器¹

可编程逻辑阵列(PLA)

可达512KB的外部存储器接口¹

电源

额定工作电压：3 V

主动模式：11 mA (5 MHz)、40 mA (41.78 MHz)

封装和温度范围

从40引脚6mm x 6mm LFCSP封装到80引脚LQFP封装¹

额定工作温度范围：-40°C至+125°C

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

功能框图

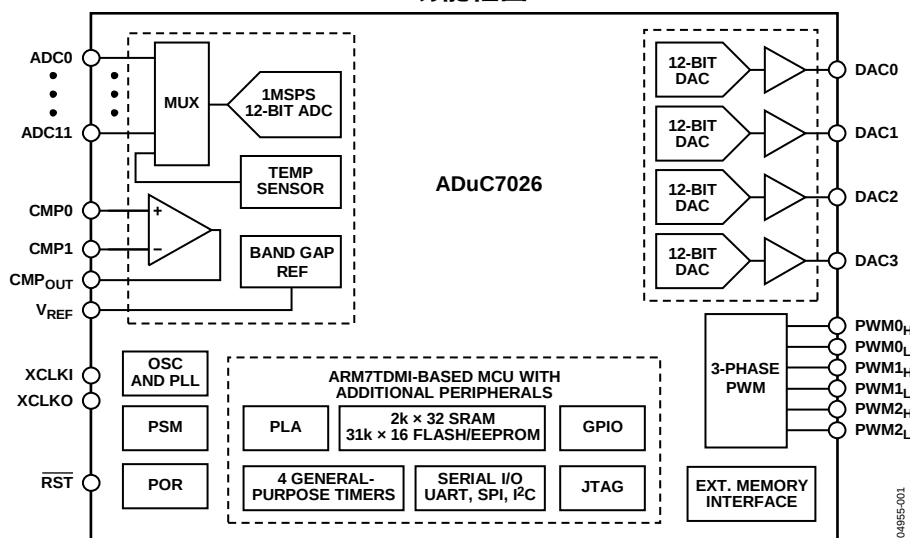


图1.

¹ 取决于具体器件型号。更多信息请参阅订购指南。

Rev. E

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

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com
Fax: 781.461.3113 ©2005-2012 Analog Devices, Inc. All rights reserved.

目录

特性.....	1	校准	46
应用.....	1	温度传感器	46
功能框图.....	1	带隙基准电压	46
修订历史.....	3	非易失Flash/EE存储器	47
概述.....	4	编程	47
详细框图.....	5	安全性	48
技术规格.....	6	Flash/EE控制接口	48
时序规格.....	9	SRAM和Flash/EE访问时间	50
绝对最大额定值.....	16	复位和重映射	50
ESD警告.....	16	其他模拟外设	52
引脚配置和功能描述.....	17	DAC	52
ADuC7019/ADuC7020/ADuC7021/ADuC7022	17	电源监控器	53
ADuC7024/ADuC7025	21	比较器	53
ADuC7026/ADuC7027	24	振荡器和锁相环—电源控制	54
ADuC7028	27	数字外设.....	57
ADuC7029	29	三相脉宽调制(PWM)	57
典型性能参数.....	31	PWM模块说明	58
术语.....	34	通用输入/输出	63
ADC技术规格	34	串口多路复用器.....	65
DAC技术规格.....	34	UART串行接口.....	65
ARM7TDMI内核概览.....	35	串行外设接口	69
Thumb模式(T)	35	I ² C兼容接口	71
长乘(M)	35	可编程逻辑阵列(PLA)	75
嵌入式ICE (I)	35	处理器基准外设.....	78
异常	35	中断系统.....	78
ARM寄存器	35	定时器	79
中断延迟.....	36	外部存储器接口.....	84
存储器结构.....	37	硬件设计考虑.....	88
存储器访问	37	电源	88
Flash/EE存储器	37	接地和电路板布局建议	89
SRAM.....	37	时钟振荡器	89
存储器映射寄存器	37	上电复位操作.....	90
ADC电路概览.....	41	典型系统配置.....	90
传递函数.....	41	开发工具.....	91
典型操作.....	42	基于PC工具	91
MMR接口	42	在线串行下载器.....	91
转换器操作	44	外形尺寸.....	92
驱动模拟输入.....	45	订购指南.....	95

修订历史 HISTORY

2012年7月—修订版D至修订版E

涉及SPI时钟时，SCLOCK更改为SCLK；涉及SPI MISO时，SPIMISO更改为MISO；涉及SPI MOSI时，SPIMOSI更改为MOSI；涉及SPI片选时，SPICSL更改为 $\overline{CS}$	通篇
更改表4、表5和图5	11
更改表6的尾注1和图6	12
更改表7和图7	13
更改表8和图8	14
更改表9和图9	15
更改图12的EPAD注释和表11	18
更改图13的EPAD注释和表12	21
更改表18的位6	43
更改“源代码示例(外部晶体选择)”部分和“源代码示例(外部时钟选择)”部分	55
更改“串行外设接口”部分	69
更改表123中的SPICON[10]和SPICON[9]描述	70
更改定时器间隔递减公式，增加定时器间隔递增公式	79
增加“时:分钟:秒:1/128格式”部分	80
更改表189	84
删除CP-40-10封装	92
更改订购指南	96

2011年5月—修订版C至修订版D

更改表4	11
更改表105	67
更新外形尺寸	91
更改订购指南	94

2009年12月—修订版B至修订版C

增加ADuC7029器件	通篇
增加表格数量并重新排序	通篇
更改图号	通篇
更改表1	6
更改图3	9
更改表3和图4	10
更改表10	16
更改图55	53
更改“串行外设接口”部分	69
更改表137	73

更改图71和图72	85
更改图73和图74	86
更新外形尺寸	91
更改订购指南	94

2007年3月—修订版A至修订版B

增加ADuC7028器件	通篇
格式更新	通篇
更改图2	5
更改表1	6
更改ADuC7026/ADuC7027部分	23
更改图21	28
更改图32说明	30
更改表14	35
更改“ADC电路概述”部分	38
更改编程部分	44
更改Flash/EE控制接口部分	45
更改表24	47
更改“RSTCLR寄存器”部分	48
更改图52	49
更改图53	50
更改比较器部分	50
更改“振荡器和锁相环—电源控制”部分	51
更改数字外设部分	54
更改中断系统部分	75
更改定时器部分	76
更改外部存储器接口部分	80
增加IOV _{DD} 电源灵敏度部分	84
更改订购指南	90

2006年1月—修订版0至修订版A

更改表1	6
增加“Flash/EE存储器可靠性”部分	43
更改表30	52
更改“串行外设接口”部分	66
更改订购指南	90

2005年10月—修订版0：初始版

ADuC7019/20/21/22/24/25/26/27/28/29

概述

ADuC7019/20/21/22/24/25/26/27/28/29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。4路DAC输出只是在特定型号上提供(ADuC7020、ADuC7026、ADuC7028和ADuC7029)。然而，在无DAC输出的情况下，这些引脚仍然可以用作ADC的输入引脚，这样ADC的输入最多可以达到16通道。ADC可以在单端或差分输入模式下工作。ADC输入电压范围为0 V至 V_{REF} 。低漂移带隙基准电压源、温度传感器和电压比较器完善了ADC的外设设置。

根据器件型号不同，片内最多可内置4个缓冲电压输出DAC。通过编程可以将DAC输出范围设置为三种电压范围之一。

这些器件通过一个片内振荡器和锁相环(PLL)产生41.78 MHz的内部高频时钟信号(UCLK)。该时钟信号通过一个可编程

时钟分频器进行中继，在其中产生MCU内核时钟工作频率。微控制器内核为ARM7TDMI®，它是一个16位/32位RISC机器，峰值性能最高可达41 MIPS。片内集成有8 KB SRAM和62 KB非易失性Flash/EE存储器。ARM7TDMI内核将所有存储器和寄存器视为一个线性阵列。

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

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。ADuC7019/ADuC7020/ADuC7021/ADuC7022/ADuC7024/ADuC7025/ADuC7026/ADuC7027/ADuC7028/ADuC7029有多种存储器类型和封装形式(详见订购指南)。

详细框图

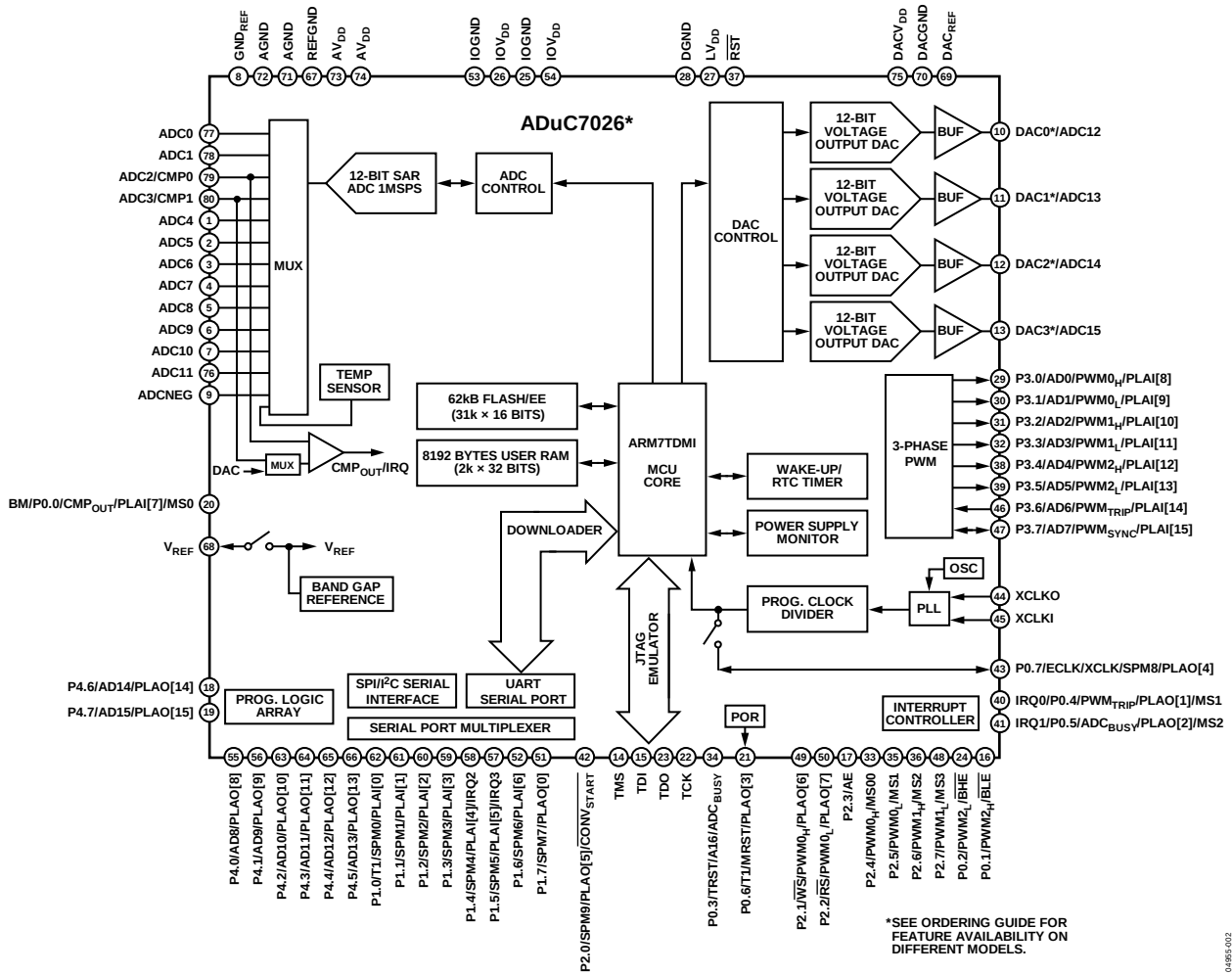


图2.

ADuC7019/20/21/22/24/25/26/27/28/29

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}	12			Bits	
分辨率				LSB	2.5 V内部基准电压
积分非线性		± 0.6	± 1.5	LSB	1.0 V外部基准电压
		± 1.0		LSB	2.5 V内部基准电压
微分非线性 ^{3,4}		± 0.5	$+1/-0.9$	LSB	1.0 V外部基准电压
		$+0.7/-0.6$		LSB	ADC输入为直流电压
直流代码分布		1		LSB	
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， $f_{SAMPLE} = 1\text{ MSPS}$
信噪比(SNR)		69		dB	包括失真和噪声成分
总谐波失真(THD)		-78		dB	
峰值谐波或杂散噪声(PHSN)		-75		dB	
通道间串扰		-80		dB	相邻通道测量
模拟输入					
输入电压范围					
差分模式			$V_{CM}^6 \pm V_{REF}/2$	V	
单端模式			0至 V_{REF}	V	
漏电流		± 1	± 6	μA	
输入电容		20		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		± 40		ppm/ $^{\circ}\text{C}$	
电源抑制比		75		dB	
输出阻抗		70		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		Bits	
相对精度		± 2		LSB	
差分非线性			± 1	LSB	保证单调性
失调误差			± 15	mV	2.5 V内部基准电压
增益误差 ⁸			± 1	%	
增益误差失配		0.1		%	DAC0满量程的百分比
模拟输出					
输出电压范围_0		0至 DAC_{REF}		V	DAC_{REF} 范围： $DACGND$ 至 $DACV_{DD}$
输出电压范围_1		0至2.5		V	
输出电压范围_2		0至 $DACV_{DD}$		V	
输出阻抗		2		Ω	

ADuC7019/20/21/22/24/25/26/27/28/29

参数	最小值	典型值	最大值	单位	测试条件/注释
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		± 20		nV-sec	
比较器					
输入失调电压		± 15		mV	迟滞通过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		7		pF	
迟滞 ^{4,6}	2		15	mV	
响应时间		3		μs	
温度传感器					
25°C时电压输出		780		mV	
电压TC		-1.3		mV/°C	
精度		± 3		°C	
电源监控器(PSM)					
IOV _{DD} 跳变点选择		2.79		V	两个可选择跳变点
		3.07		V	
电源跳变点精度		± 2.5		%	已选跳变点标称电压
上电复位		2.36		V	
复位引脚干扰抑制 ³		50		μs	
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	$T_J = 85^\circ\text{C}$
数据保持 ¹⁰	20			年	
数字输入					除XCLKI和XCLKO外的所有数字输入
逻辑1输入电流	± 0.2		± 1	μA	$V_{IH} = IOV_{DD}$ 或 $V_{IH} = 5\text{ V}$ $V_{IL} = 0\text{ V}$; ADuC7019/20/21/22/24/25/29的TDI除外 $V_{IL} = 0\text{ V}$; ADuC7019/20/21/22/24/25/29的TDI
逻辑0输入电流	-40		-60	μA	
	-80		-120	μA	
输入电容	10			pF	
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	$I_{SOURCE} = 1.6\text{ mA}$ $I_{SINK} = 1.6\text{ mA}$
输出低电压 V_{OL} ¹¹			0.4	V	
晶体输入XCLKI和XCLKO					
逻辑输入, 仅限XCLKI					
输入低电压 V_{INL}		1.1		V	
输入高电压 V_{INH}		1.7		V	
XCLKI输入电容		20		pF	
XCLKO输出电容		20		pF	
内部振荡器		32.768		kHz	$T_A = 0^\circ\text{C}$ 至 85°C 范围
			± 3	%	
			$\pm 2^4$	%	

ADuC7019/20/21/22/24/25/26/27/28/29

参数	最小值	典型值	最大值	单位	测试条件/注释
MCU时钟速率					
采用32 kHz内部振荡器		326		kHz	CD ¹² = 7
采用32 kHz外部晶体		41.78		MHz	CD ¹² = 0
使用外部时钟	0.05		44	MHz	T _A = 85°C
	0.05		41.78	MHz	T _A = 125°C
启动时间					内核时钟= 41.78 MHz
上电时		130		ms	
从暂停/休眠模式		24		ns	CD ¹² = 0
		3.06		μs	CD ¹² = 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 ^{13, 14}					
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		200		μA	ADC在空闲模式；除ADuC7019外所有器件
		400		μA	ADC在空闲模式；只有ADuC7019
DACV _{DD} 电流 ¹⁵		3	25	μA	
数字电源电流					
IOV _{DD} 正常模式下电流					从Flash/EE执行代码
		7	10	mA	CD ¹² = 7
		11	15	mA	CD ¹² = 3
		40	45	mA	CD ¹² = 0 (41.78 MHz时钟)
IOV _{DD} 暂停模式下电流		25	30	mA	CD ¹² = 0 (41.78 MHz时钟)
IOV _{DD} 休眠模式下电流		250	400	μA	T _A = 85°C
		600	1000	μA	T _A = 125°C
附加电源电流					
ADC		2		mA	@ 1 MSPS
		0.7		mA	@ 62.5 kSPS
DAC		700		μA	每DAC
ESD测试					2.5 V基准电压, T _A = 25°C
最大HBM通过电压			4	kV	
最大FCIDM通过电压			0.5	kV	

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

² 适用于所有ADC输入通道。

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

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49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来满足规格要求(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2.5 V基准电压)计算出来的。

⁹ 耐久性是在-40°C、+25°C、+85°C及+125°C时依据JEDEC 22标准方法A117来认定的。

¹⁰ 根据JEDEC 22m标准方法A117，保持期限相当于85°C结温时的寿命。保持期限会随着结温递减。

¹¹ 测试是在最多8个I/O端口输出低电平时进行的。

¹² 参见POWCON寄存器。

¹³ 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3.6 V、暂停模式供电电压为3.6 V、休眠模式供电电压为3.6 V。

¹⁴ 在一个Flash/EE擦写周期中，IOV_{DD}电源电流通常降低2 mA。

¹⁵ 对于ADuC7019/20/21/22，该电流必须加上AV_{DD}的电流。

时序规格

表2. 外部存储器写周期

参数	最小值	典型值	最大值	单位
CLK ¹		UCLK		
t _{MS_AFTER_CLKH}	0		4	ns
t _{ADDR_AFTER_CLKH}	4		8	ns
t _{AE_H_AFTER_MS}		½ CLK		
t _{AE}		(XMxPAR[14:12] + 1) × CLK		
t _{HOLD_ADDR_AFTER_AE_L}		½ CLK + (!XMxPAR[10]) × CLK		
t _{HOLD_ADDR_BEFORE_WR_L}		(!XMxPAR[8]) × CLK		
t _{WR_L_AFTER_AE_L}		½ CLK + (!XMxPAR[10] + !XMxPAR[8]) × CLK		
t _{DATA_AFTER_WR_L}	8		12	ns
t _{WR}		(XMxPAR[7:4] + 1) × CLK		
t _{WR_H_AFTER_CLKH}	0		4	ns
t _{HOLD_DATA_AFTER_WR_H}		(!XMxPAR[8]) × CLK		
t _{BEN_AFTER_AE_L}		½ CLK		
t _{RELEASE_MS_AFTER_WR_H}		(!XMxPAR[8] + 1) × CLK		

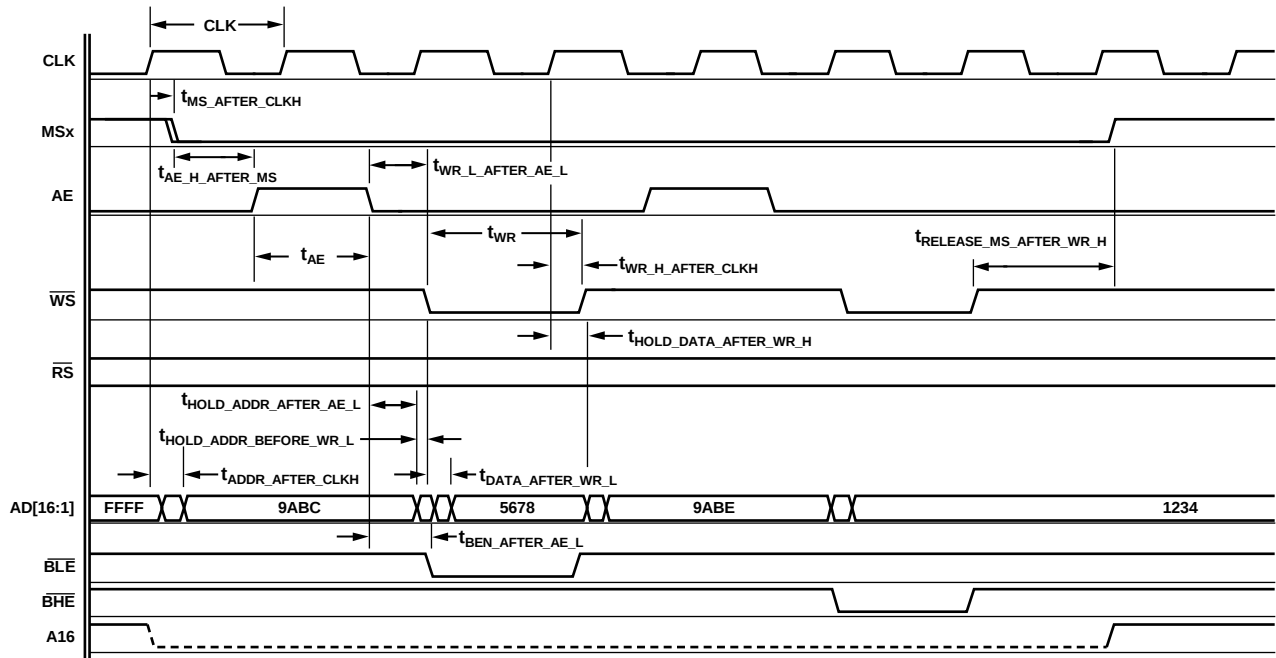
¹ 参见表78。

图3. 外部存储器写周期(参见表78)

表3. 外部存储器读周期

参数	最小值	典型值	最大值	单位
CLK ¹	1/MD 时钟	ns typ × (POWCON[2:0] + 1)		
t _{MS_AFTER_CLKH}	4		8	ns
t _{ADDR_AFTER_CLKH}	4		16	ns
t _{AE_H_AFTER_MS}		½ CLK		
t _{AE}		(XMxPAR[14:12] + 1) × CLK		
t _{HOLD_ADDR_AFTER_AE_L}		½ CLK + (! XMxPAR[10]) × CLK		
t _{RD_L_AFTER_AE_L}		½ CLK + (! XMxPAR[10] + ! XMxPAR[9]) × CLK		
t _{RD_H_AFTER_CLKH}	0		4	
t _{RD}		(XMxPAR[3:0] + 1) × CLK		
t _{DATA_BEFORE_RD_H}	16			ns
t _{DATA_AFTER_RD_H}	8	+ (! XMxPAR[9]) × CLK		
t _{RELEASE_MS_AFTER_RD_H}		1 × CLK		

¹ 参见表78。

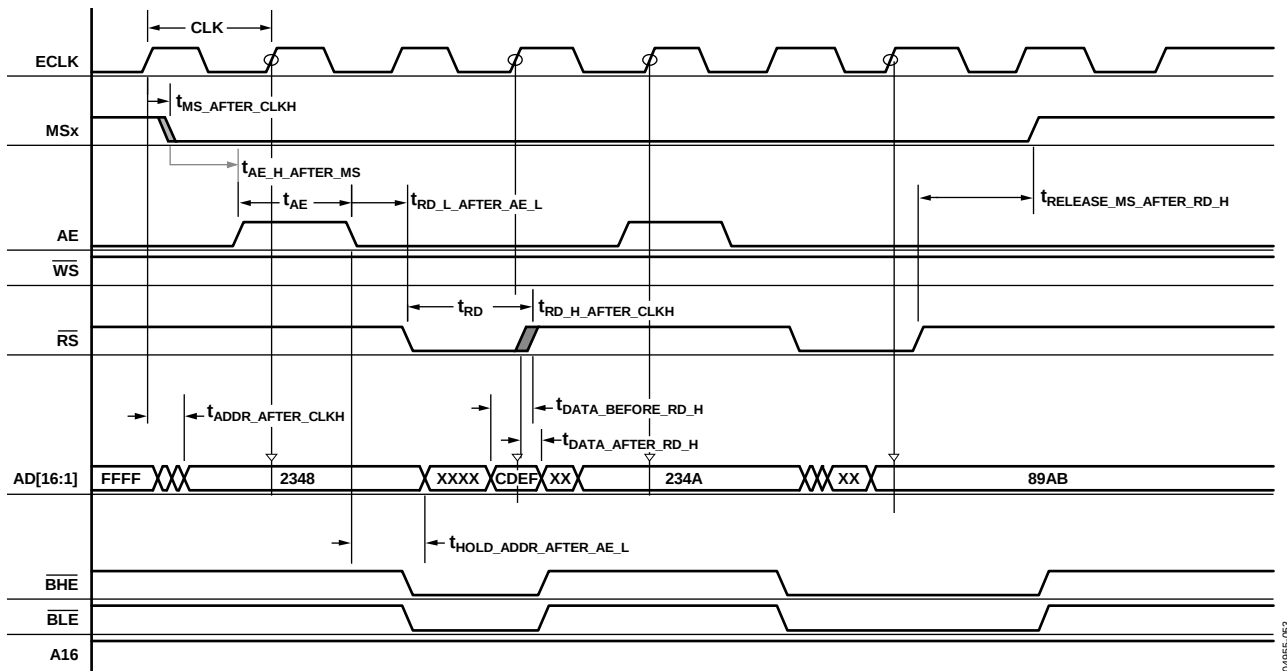


图4. 外部存储器读周期(参见表78)

04955-053

表4. 快速模式下I²C时序(400 kHz)

参数	描述	从器件		主机	单位
		最小值	最大值	典型值	
t _L	SCL低电平脉宽 ¹	200		1360	ns
t _H	SCL高电平脉宽 ¹	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		400	ns
t _{BUF}	一个结束条件和起始条件之间的总线空闲时间	1.3			ns
t _R	SCL和SDA的上升时间		300	200	ns
t _F	SCL和SDA的下降时间		300		ns
t _{SUP}	尖峰抑制脉宽		50		ns

¹ t_{HCLK}取决于时钟分频器或POWCON MMR的CD位。t_{HCLK} = t_{UCLK}/2^{CD}；参见图57。

表5. 标准模式下I²C时序(100 kHz)

参数	描述	从器件		主机	单位
		最小值	最大值	典型值	
t _L	SCL低电平脉宽 ¹	4.7			μs
t _H	SCL高电平脉宽 ¹	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

¹ t_{HCLK}取决于时钟分频器或POWCON MMR的CD位。t_{HCLK} = t_{UCLK}/2^{CD}；参见图57。

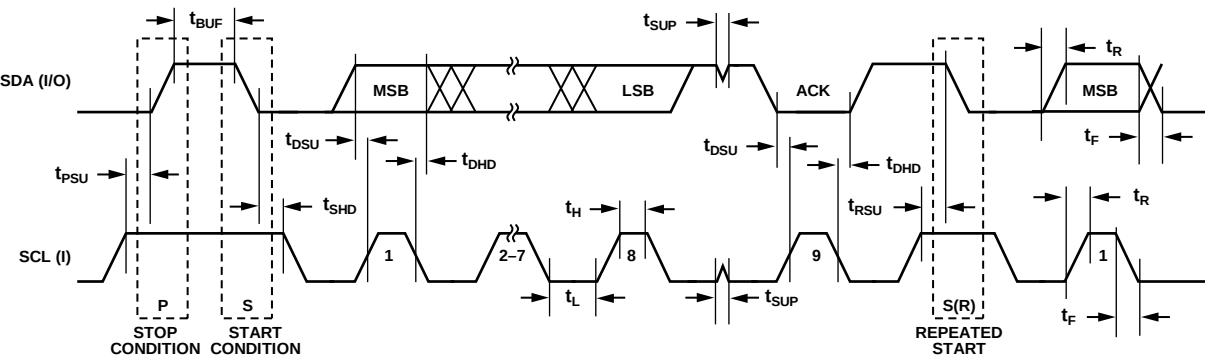


图 5. I²C兼容接口时序

04955-054

表6. 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		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK高电平脉宽 ¹		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{DAV}	SCLK边沿之后数据输出有效时间			25	ns
t_{DSU}	SCLK边沿之前数据输入建立时间 ²	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK边沿之后数据输入保持时间 ²	$2 \times t_{UCLK}$			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_{HCLK} 取决于时钟分频器或POWCON MMR的CD位。 $t_{HCLK} = t_{UCLK} / 2^{CD}$ ；参见图57。

² $t_{UCLK} = 23.9$ ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟；参见图57。

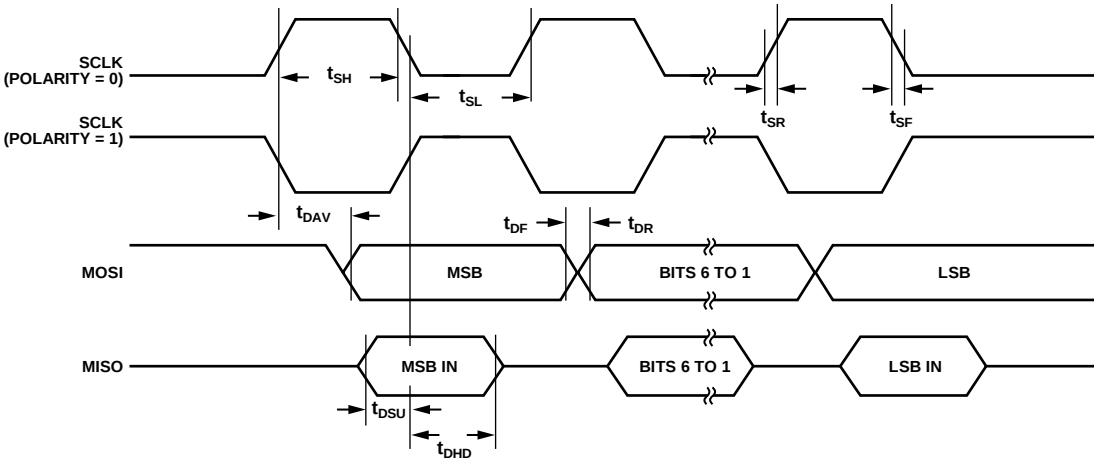


图6. SPI主机定时(相位模式 = 1)

表7. 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		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK高电平脉宽 ¹		$(SPIDIV + 1) \times t_{HCLK}$		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 ²	$1 \times t_{UCLK}$			ns
t_{DHD}	SCLK边沿之后数据输入保持时间 ²	$2 \times t_{UCLK}$			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_{HCLK} 取决于时钟分频器或POWCON MMR的CD位。 $t_{HCLK} = t_{UCLK} / 2^{CD}$ ；参见图57。

² $t_{UCLK} = 23.9$ ns。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟；参见图57。

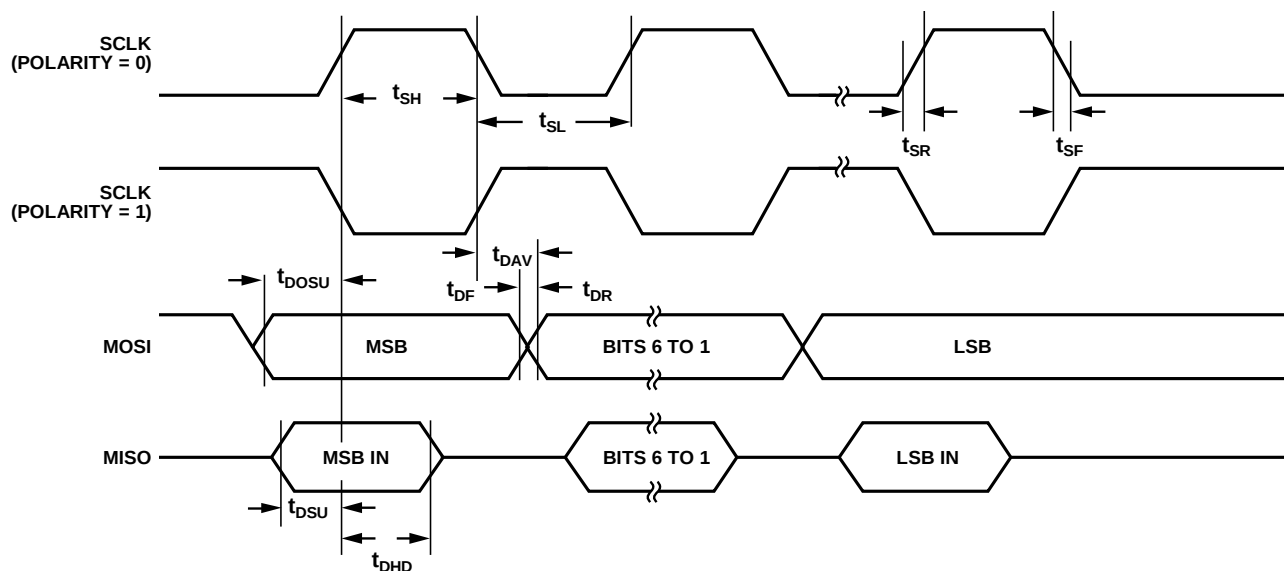


图7. SPI主机定时(相位模式 = 0)

04955-036

表8. SPI从机定时(相位模式 = 1)

参数	描述	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	$\overline{CS}$ 到SCLK边沿 ¹	$(2 \times t_{HCLK}) + (2 \times t_{UCLK})$			ns
t_{SL}	SCLK低电平脉宽 ²		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK高电平脉宽 ²		$(SPIDIV + 1) \times t_{HCLK}$		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
t_{SFS}	$\overline{CS}$ 在SCLK沿后变高	0			ns

¹ $t_{UCLK} = 23.9 \text{ ns}$ 。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟；参见图57。

² t_{HCLK} 取决于时钟分频器或POWCON MMR的CD位。 $t_{HCLK} = t_{UCLK} / 2^{CD}$ ；参见图57。

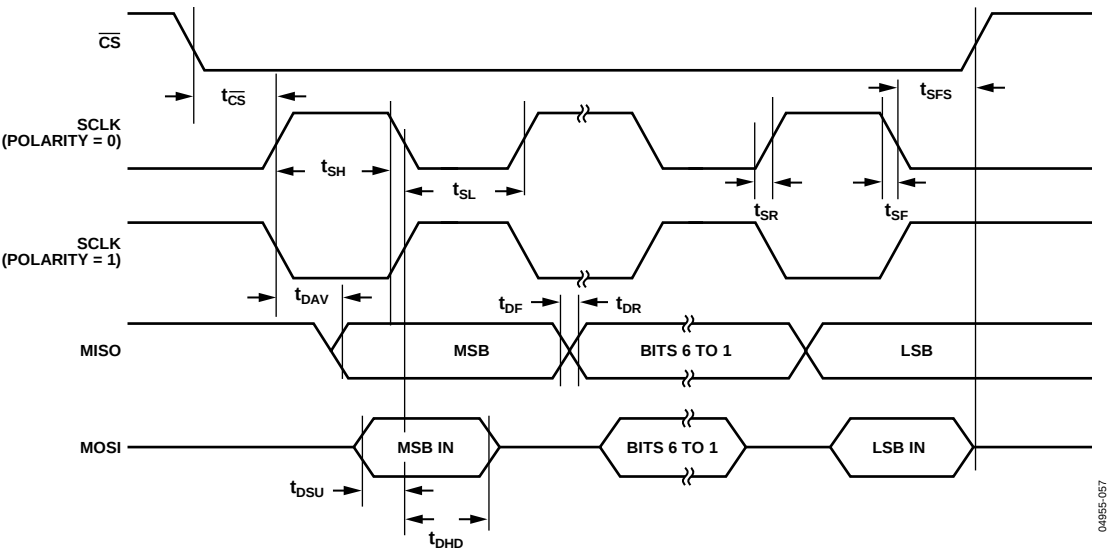


图8. SPI从机定时(相位模式 = 1)

表9. SPI从机定时(相位模式 = 0)

参数	描述	最小值	典型值	最大值	单位
$t_{\overline{CS}}$	$\overline{CS}$ 到SCLK边沿 ¹	$(2 \times t_{HCLK}) + (2 \times t_{UCLK})$			ns
t_{SL}	SCLK低电平脉宽 ²		$(SPIDIV + 1) \times t_{HCLK}$		ns
t_{SH}	SCLK高电平脉宽 ²		$(SPIDIV + 1) \times t_{HCLK}$		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
t_{DOCS}	$\overline{CS}$ 边沿之后数据输出有效			25	ns
t_{SFS}	$\overline{CS}$ 在SCLK沿后变高	0			ns

¹ $t_{UCLK} = 23.9 \text{ ns}$ 。其对应于在时钟分频器之前来自PLL的41.78 MHz内部时钟；参见图57。

² t_{HCLK} 取决于时钟分频器或POWCON MMR的CD位。 $t_{HCLK} = t_{UCLK} / 2^{CD}$ ；参见图57。

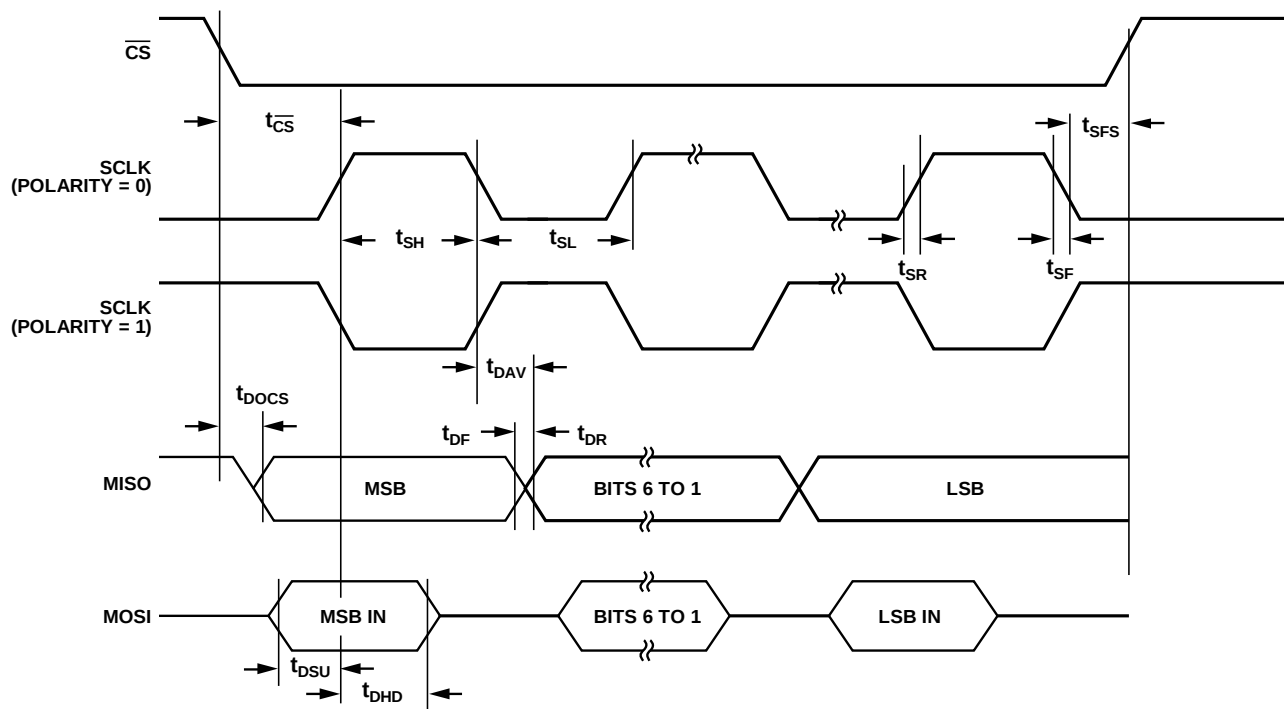


图9. SPI从机定时(相位模式 = 0)

04955-058

绝对最大额定值

除非另有说明，AGND = REFGND = DACGND = GND_{REF}，T_A = 25°C。

表10.

参数	额定值
AV _{DD} 至IOV _{DD}	−0.3 V至+0.3 V
AGND至DGND	−0.3 V至+0.3 V
IOV _{DD} 至IOGND，AV _{DD} 至AGND	−0.3 V至+6 V
数字输入电压至IOGND	−0.3 V至+5.3 V
数字输出电压至IOGND	−0.3 V至IOV _{DD} + 0.3 V
V _{REF} 至AGND	−0.3 V至AV _{DD} + 0.3 V
模拟输入至AGND	−0.3 V至AV _{DD} + 0.3 V
模拟输出至AGND	−0.3 V至AV _{DD} + 0.3 V
工业温度范围	−40°C至+125°C
存储温度范围	−65°C至+150°C
结温	150°C
θ _{JA} 热阻	
40引脚LFCSP	26°C/W
49引脚CSP_BGA	80°C/W
64引脚LFCSP	24°C/W
64引脚CSP_BGA	75°C/W
64引脚LQFP	47°C/W
80引脚LQFP	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。

引脚配置和功能描述

ADuC7019/ADuC7020/ADuC7021/ADuC7022

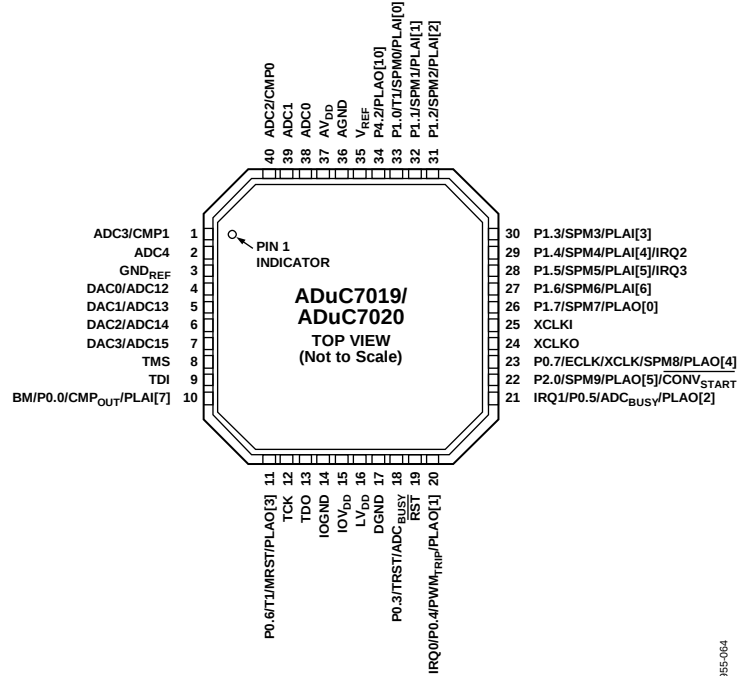


图10. ADuC7019/ADuC7020 40引脚LFCSP_VQ封装引脚配置

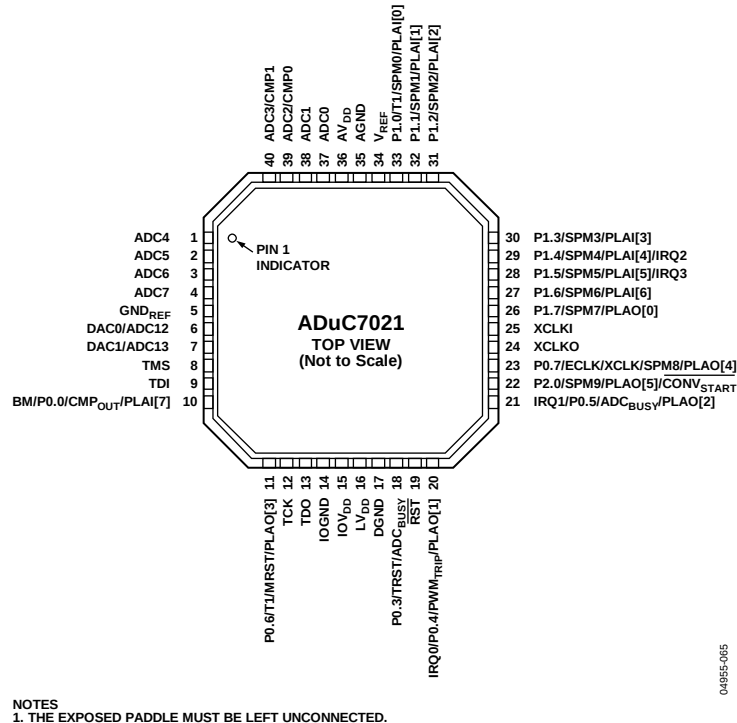
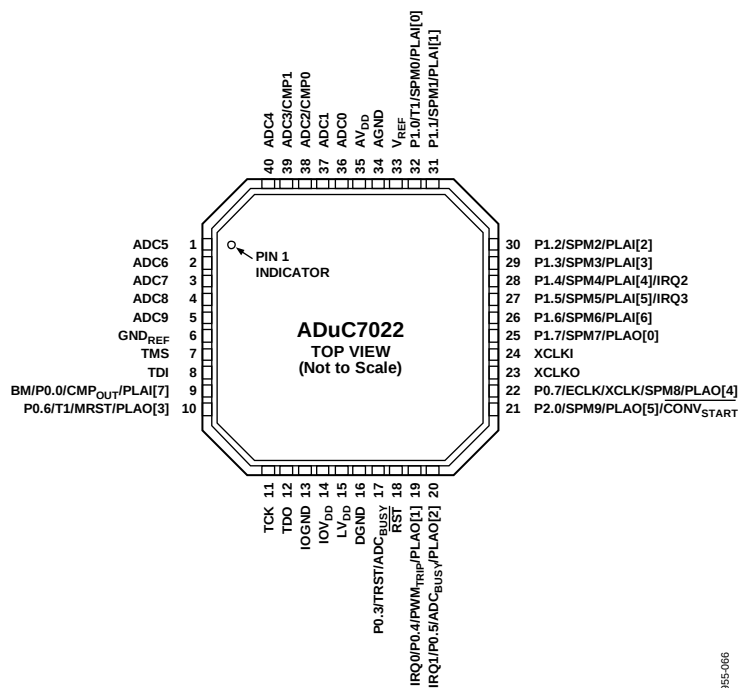


图 11. ADuC7021 40引脚LFCSP_VQ封装引脚配置

ADuC7019/20/21/22/24/25/26/27/28/29



NOTES
1. THE EXPOSED PADDLE MUST BE SOLDERED AND EITHER CONNECTED TO AGND OR LEFT FLOATING.

图 12. ADuC7022 40引脚LFCSP_VQ封装引脚配置

表11. ADuC7019/ADuC7020/ADuC7021/ADuC7022引脚功能描述

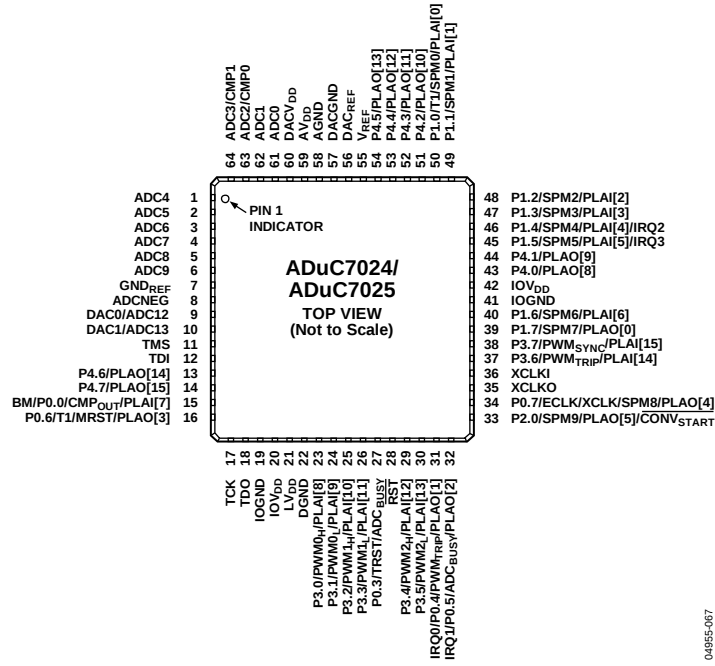
引脚编号			引脚名称	描述
7019/7020	7021	7022		
38	37	36	ADC0	单端或差分模拟输入0。
39	38	37	ADC1	单端或差分模拟输入1。
40	39	38	ADC2/CMP0	单端或差分模拟输入2/比较器正相输入。
1	40	39	ADC3/CMP1	单端或差分模拟输入3(ADuC7019缓冲输入)/比较器反相输入端。
2	1	40	ADC4	单端或差分模拟输入4。
–	2	1	ADC5	单端或差分模拟输入5。
–	3	2	ADC6	单端或差分模拟输入6。
–	4	3	ADC7	单端或差分模拟输入7。
–	–	4	ADC8	单端或差分模拟输入8。
–	–	5	ADC9	单端或差分模拟输入9。
3	5	6	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。
4	6	–	DAC0/ADC12	DAC0 电压输出/单端或差分模拟输入12。
5	7	–	DAC1/ADC13	DAC1电压输出/单端或差分模拟输入 13。
6	–	–	DAC2/ADC14	DAC2电压输出/单端或差分模拟输入14。
7	–	–	DAC3/ADC15	ADuC7020 DAC3电压输出。在ADuC7019芯片内，必须在该引脚和AGND/单端或差分模拟输入15之间连接一个10 nF电容(参见图43)。
8	8	7	TMS	测试模式选择，JTAG测试端口输入。调试和下载访问。 此引脚具有一个连接至IOV _{DD} 的内部上拉电阻。在有些情况下，还需要一个外部上拉电阻(约100K)，以确保器件不会进入错误状态。
9	9	8	TDI	测试数据输入，JTAG测试端口输入。调试和下载访问。

引脚编号			引脚名称	描述
7019/7020	7021	7022		
10	10	9	BM/P0.0/CMP _{OUT} /PLAI[7]	多功能输入输出引脚。引导模式(BM)。BM(复位时如果BM为低电平, 则ADuC7019/20/21/22进入串行下载模式; 如果BM由1 kΩ电阻上拉至高电平, 则执行代码)/通用输入输出端口P0.0/电压比较器输出/PLA输入单元7。
11	11	10	P0.6/T1/MRST/PLAO[3]	多功能引脚。复位后拉低。通用输入输出端口P 0.6/定时器1输入/上电复位输出/可编程逻辑阵列输出单元3。
12	12	11	TCK	测试时钟,JTAG测试端口输入。调试和下载访问。此引脚具有一个连接至IOV _{DD} 的内部上拉电阻。在有些情况下, 还需要一个外部上拉电阻(约100K), 以确保器件不会进入错误状态。
13	13	12	TDO	测试数据输出, JTAG测试端口输出。调试和下载访问。
14	14	13	IOGND	GPIO地(参见表78)。通常连接到DGND。
15	15	14	IOV _{DD}	3.3 V电源, 用于GPIO(参见表78)和片内稳压器输入。
16	16	15	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47 uF电容器连接至DGND。
17	17	16	DGND	内核逻辑地。
18	18	17	P0.3/TRST/ADC _{BUSY}	通用输入输出P0.3/测试复位, JTAG测试端口输入/ADC _{BUSY} 信号输出。
19	19	18	$\overline{\text{RST}}$	复位输入, 低电平有效。
20	20	19	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/通用输入输出端口P 0.4/PWM触发外部输入/可编程逻辑阵列输出单元1。
21	21	20	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/通用输入输出端口P0.5/ADC _{BUSY} 信号输出/可编程逻辑阵列输出单元2。
22	22	21	P2.0/SPM9/PLAO[5]/ $\overline{\text{CONV}}_{\text{START}}$	串行复用端口。通用输入输出端口P2.0/UART/可编程逻辑阵列输出单元5/ADC起始转换输入信号。
23	23	22	P0.7/ECLK/XCLK/SPM8/PLAO[4]	串行复用端口。通用输入输出端口P0.7/外部时钟信号输出/内部时钟发生器电路输入/UART/可编程逻辑阵列输出单元4。
24	24	23	XCLKO	晶体振荡反相器输出。
25	25	24	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。
26	26	25	P1.7/SPM7/PLAO[0]	串行复用端口。通用输入输出端口P 1.7/UART, SPI/可编程逻辑阵列输出单元0。
27	27	26	P1.6/SPM6/PLAI[6]	串行复用端口。通用输入输出端口P 1.6/UART, SPI/可编程逻辑阵列输入单元6。
28	28	27	P1.5/SPM5/PLAI[5]/IRQ3	串行复用端口。通用输入输出端口P 1.5/UART, SPI/可编程逻辑阵列输入单元5/外部中断请求3, 高电平有效。
29	29	28	P1.4/SPM4/PLAI[4]/IRQ2	串行复用端口。通用输入输出端口P 1.4/UART, SPI/可编程逻辑阵列输入单元4/外部中断请求2, 高电平有效。
30	30	29	P1.3/SPM3/PLAI[3]	串行复用端口。通用输入输出端口P 1.3/UART, I2C1/可编程逻辑阵列输入单元3。
31	31	30	P1.2/SPM2/PLAI[2]	串行复用端口。通用输入输出端口P 1.2/UART, I2C1/可编程逻辑阵列输入单元2。
32	32	31	P1.1/SPM1/PLAI[1]	串行复用端口。通用输入输出端口P 1.1/UART, I2C0/可编程逻辑阵列输入单元1。
33	33	32	P1.0/T1/SPM0/PLAI[0]	串行复用端口。通用输入输出端口P 1.0/定时器1输入/UART, I2C0/可编程逻辑阵列输入单元0。
34	–	–	P4.2/PLAO[10]	通用输入输出端口P4.2/可编程逻辑阵列输出单元10。

ADuC7019/20/21/22/24/25/26/27/28/29

引脚编号			引脚名称	描述
7019/7020	7021	7022		
35	34	33	V _{REF}	2.5 V内部基准电压。在使用内部基准电压源时必须连接至一个0.47 μF电容。
36	35	34	AGND	模拟地。模拟电路的地基准点。
37	36	35	AV _{DD}	3.3 V模拟电源。
0	0	0	EP	裸露焊盘。ADuC7019/ADuC7020/ADuC7021/ADuC7022的引脚配置有一个裸露焊盘，必须将其焊接并连接到AGND或悬空。

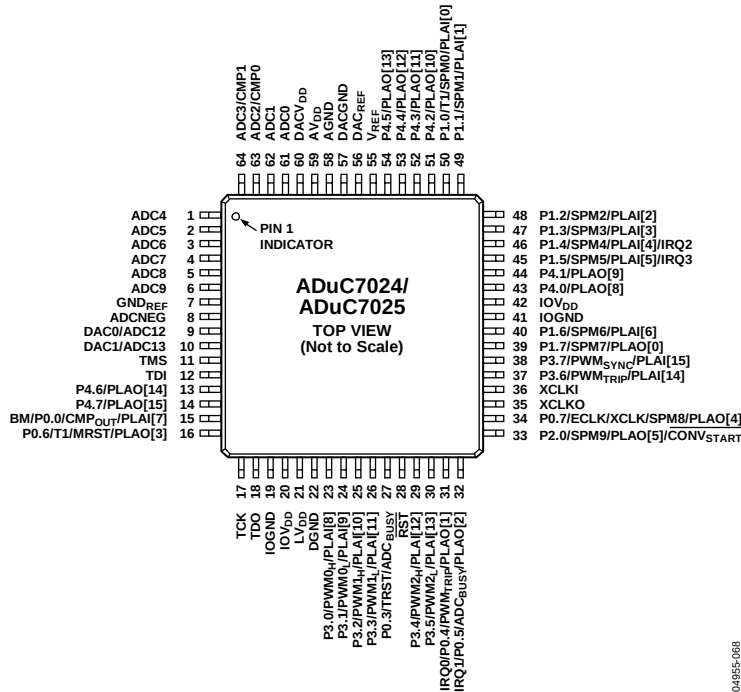
ADuC7024/ADuC7025



04955-067

NOTES
1. THE EXPOSED PADDLE MUST BE SOLDERED AND EITHER CONNECTED TO AGND OR LEFT FLOATING.

图13. ADuC7024/ADuC7025 64引脚LFCSP_VQ封装引脚配置



04955-068

Figure 14. ADuC7024/ADuC7025 64引脚LQFP封装引脚配置

ADuC7019/20/21/22/24/25/26/27/28/29

表12. 引脚功能描述(ADuC7024/ADuC7025 64引脚LFCSP_VQ封装和64引脚LQFP封装)

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	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。
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0 V至1 V之间。
9	DAC0/ADC12	DAC0电压输出/单端或差分模拟输入12。在ADuC7025上没有DAC输出。
10	DAC1/ADC13	DAC1电压输出/单端或差分模拟输入13。在ADuC7025上没有DAC输出。
11	TMS	JTAG测试端口输入，测试模式选择。调试和下载访问。
12	TDI	JTAG 测试端口输入，测试数据输入。调试和下载访问
13	P4.6/PLAO[14]	通用输入输出端口P4.6/可编程逻辑阵列输出单元14。
14	P4.7/PLAO[15]	通用输入输出端口P4.7/可编程逻辑阵列输出单元15。
15	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ADuC7024/ADuC7025进入下载模式；如果BM通过1 kΩ电阻上拉至高电平，则执行代码)/通用输入输出端口P0.0/电压比较器输出/可编程逻辑阵列输入单元7。
16	P0.6/T1/MRST/PLAO[3]	多功能引脚，复位后输出低电平。通用输入输出端口P 0.6/定时器1输入/上电复位输出/可编程逻辑阵列输出单元3。
17	TCK	JTAG测试端口输入，测试时钟。调试和下载访问。
18	TDO	JTAG测试端口输出，测试数据输出。调试和下载访问。
19	IOGND	Ground for GPIO (see Table 78).通常连接到DGND。
20	IOV _{DD}	3.3 V电源，用于GPIO(参见表78)和片内稳压器输入。
21	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47 μF电容器连接至DGND。
22	DGND	内核逻辑地。
23	P3.0/PWM0 _H /PLAI[8]	通用输入输出端口P 3.0/PWM相位0高边输出/可编程逻辑阵列输入单元8。
24	P3.1/PWM0 _L /PLAI[9]	通用输入输出端口P 3.1/PWM相位0低边输出/可编程逻辑阵列输入单元9。
25	P3.2/PWM1 _H /PLAI[10]	通用输入输出端口P 3.2/PWM相位1高边输出/可编程逻辑阵列输入单元10。
26	P3.3/PWM1 _L /PLAI[11]	通用输入输出端口P 3.3/PWM相位1低边输出/可编程逻辑阵列输入单元11。
27	P0.3/TRST/ADC _{BUSY}	通用输入输出端口P0.3/JTAG测试端口输入，测试复位/ADC _{BUSY} 信号输出。
28	RST	复位输入，低电平有效。
29	P3.4/PWM2 _H /PLAI[12]	通用输入输出端口P 3.4/PWM相位2高边输出/可编程逻辑阵列输入单元12。
30	P3.5/PWM2 _L /PLAI[13]	通用输入输出端口P 3.5/PWM相位2低边输出/可编程逻辑阵列输入单元13。
31	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/通用输入输出端口P 0.4/PWM触发外部输入/可编程逻辑阵列输出单元1。
32	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/通用输入输出端口P 0.5/ADC _{BUSY} 信号输出/可编程逻辑阵列输出单元2。
33	P2.0/SPM9/PLAO[5]/CONV _{START}	串行复用端口。通用输入输出端口P 2.0/UART/可编程逻辑阵列输出单元5/ADC起始转换输入信号。
34	P0.7/ECLK/XCLK/SPM8/PLAO[4]	串行复用端口。通用输入输出端口P0.7/外部时钟信号输出/内部时钟发生器电路输入/UART/可编程逻辑阵列输出单元4。
35	XCLKO	晶体振荡反相器输出。
36	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。

引脚编号	引脚名称	描述
37	P3.6/PWM _{TRIP} /PLAI[14]	通用输入输出端口P 3.6/PWM安全关断/可编程逻辑阵列输入单元14。
38	P3.7/PWM _{SYNC} /PLAI[15]	通用输入输出端口P 3.7/PWM同步输入输出/可编程逻辑阵列输入单元15。
39	P1.7/SPM7/PLAO[0]	串行复用端口。通用输入输出端口P 1.7/UART, SPI/可编程逻辑阵列输出单元0。
40	P1.6/SPM6/PLAI[6]	串行复用端口。通用输入输出端口P 1.6/UART, SPI/可编程逻辑阵列输入单元6。
41	IOGND	GPIO地(参见表78)。通常连接到DGND。
42	IOV _{DD}	3.3 V电源, 用于GPIO(参见表78)和片内稳压器输入。
43	P4.0/PLAO[8]	通用输入输出端口P4.0/可编程逻辑阵列输出单元8。
44	P4.1/PLAO[9]	通用输入输出端口P4.1/可编程逻辑阵列输出单元9。
45	P1.5/SPM5/PLAI[5]/IRQ3	串行复用端口。通用输入输出端口P 1.5/UART, SPI/可编程逻辑阵列输入单元5/ 外部中断请求3, 高电平有效。
46	P1.4/SPM4/PLAI[4]/IRQ2	串行复用端口。通用输入输出端口P 1.4/UART, SPI/可编程逻辑阵列输入单元4/ 外部中断请求2, 高电平有效。
47	P1.3/SPM3/PLAI[3]	串行复用端口。通用输入输出端口P 1.3/UART, I2C1/可编程逻辑阵列输入单元3。
48	P1.2/SPM2/PLAI[2]	串行复用端口。通用输入输出端口P 1.2/UART, I2C1/可编程逻辑阵列输入单元2。
49	P1.1/SPM1/PLAI[1]	串行复用端口。通用输入输出端口P 1.1/UART, I2C0/可编程逻辑阵列输入单元1。
50	P1.0/T1/SPM0/PLAI[0]	串行复用端口。通用输入输出端口P 1.0/定时器1输入/UART, I2C0/可编程逻辑阵列 输入单元0。
51	P4.2/PLAO[10]	通用输入输出端口P4.2/可编程逻辑阵列输出单元10。
52	P4.3/PLAO[11]	通用输入输出端口P4.3/可编程逻辑阵列输出单元11。
53	P4.4/PLAO[12]	通用输入输出端口P4.4/可编程逻辑阵列输出单元12。
54	P4.5/PLAO[13]	通用输入输出端口P4.5/可编程逻辑阵列输出单元13。
55	V _{REF}	2.5 V内部基准电压。在使用内部基准电压源时必须连接至一个0.47 μF电容。
56	DAC _{REF}	DAC外部基准电压。范围: DACGND至DACV _{DD} 。
57	DACGND	DAC地。通常连接到AGND。
58	AGND	模拟地。模拟电路的地基准点。
59	AV _{DD}	3.3 V模拟电源。
60	DACV _{DD}	DAC 3.3 V电压源。必须连接到AV _{DD} 。
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/比较器正相输入。
64	ADC3/CMP1	单端或差分模拟输入3/比较器反相输入。
0	EP	裸露焊盘。ADuC7024/ADuC7025 LFCSP_VQ封装的引脚配置有一个裸露焊盘, 必须将其焊接并连接到AGND或悬空。

ADuC7019/20/21/22/24/25/26/27/28/29

ADuC7026/ADuC7027

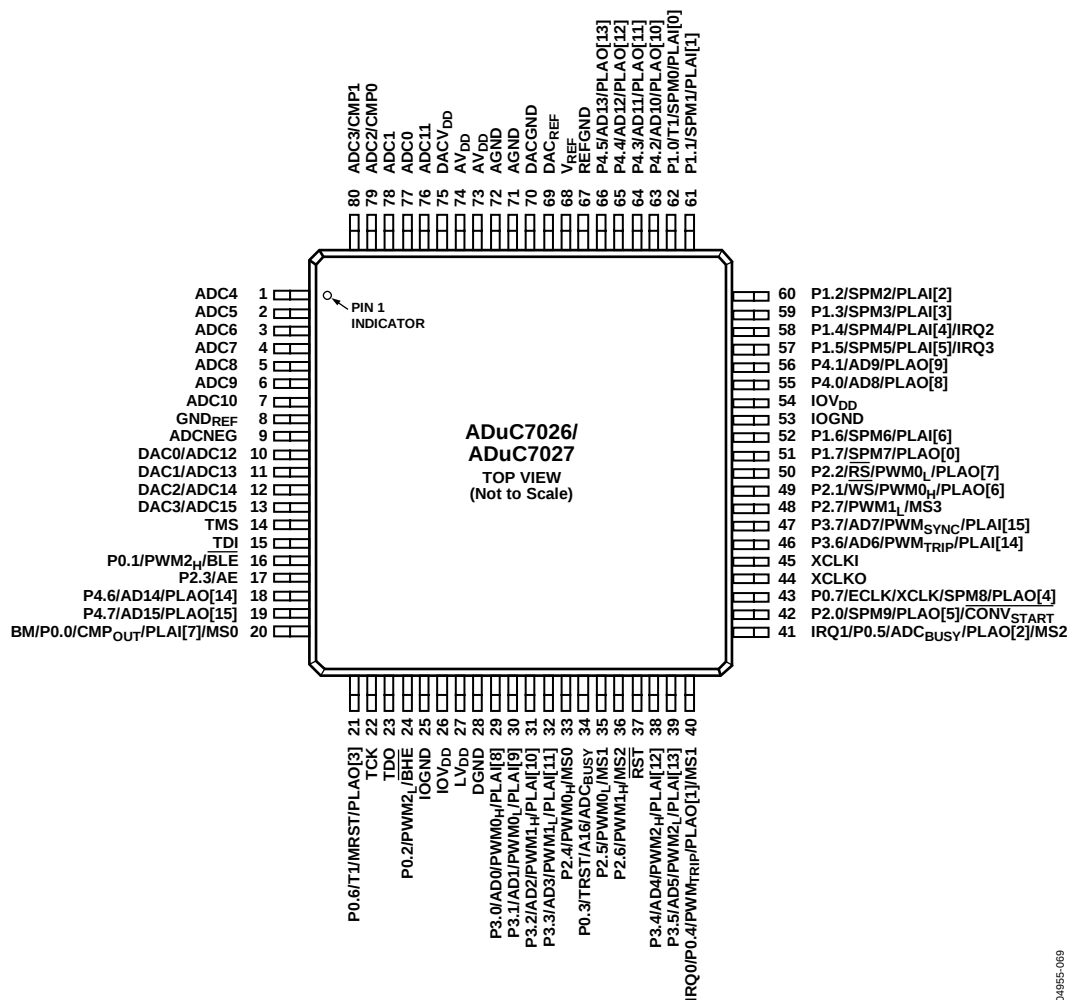


图15. 80引脚LQFP封装引脚配置(ADuC7026/ADuC7027)

表13 引脚功能描述(ADuC7026/ADuC7027)

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	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。
9	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。
10	DAC0/ADC12	DAC0模拟电压输出/单端或差分输入12。在ADuC7027上没有DAC输出。
11	DAC1/ADC13	DAC1模拟电压输出/单端或差分输入13。在ADuC7027上没有DAC输出。
12	DAC2/ADC14	DAC2模拟电压输出/单端或差分输入14。在ADuC7027上没有DAC输出。
13	DAC3/ADC15	DAC3模拟电压输出/单端或差分输入15。在ADuC7027上没有DAC输出。
14	TMS	JTAG测试端口输入，测试模式选择。调试和下载访问。

引脚编号	引脚名称	描述
15	TDI	JTAG测试端口输入，测试数据输入。调试和下载访问。
16	P0.1/PWM2 _H /BLE	通用输入输出端口P0.1/PWM相位2高边输出/外部存储器低字节使能。
17	P2.3/AE	通用输入输出端口P2.3/外部存储器访问使能。
18	P4.6/AD14/PLAO[14]	通用输入输出端口P4.6/外部存储器接口/可编程逻辑阵列输出单元14。
19	P4.7/AD15/PLAO[15]	通用输入输出端口P4.7/外部存储器接口/可编程逻辑阵列输出单元15。
20	BM/P0.0/CMP _{OUT} /PLAI[7]/MS0	多功能输入输出引脚。引导模式。BM(如果在复位时BM为低，ADuC7026/DuC7027进入UART下载模式；如果BM使用1 kΩ电阻上拉至高电平则执行代码)/通用输入输出端口P0.0/电压比较器输出/可编程逻辑阵列输入单元7/外部存储器选择0。
21	P0.6/T1/MRST/PLAO[3]	多功能引脚，复位后输出低电平。通用输入输出端口P0.6/定时器1输入/上电复位输出/可编程逻辑阵列输出单元3。
22	TCK	JTAG测试端口输入，测试时钟。调试和下载访问。
23	TDO	JTAG测试端口输出，测试数据输出。调试和下载访问。
24	P0.2/PWM2 _L /BHE	通用输入输出端口P0.2/PWM相位2低边输出/外部存储器高字节使能。
25	IOGND	GPIO地(参见表78)。通常连接到DGND。
26	IOV _{DD}	3.3 V电源，用于GPIO(参见表78)和片内稳压器输入。
27	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47 μF电容器连接至DGND。
28	DGND	内核逻辑地。
29	P3.0/AD0/PWM0 _H /PLAI[8]	通用输入输出端口P3.0/外部存储器接口/PWM相位0高边输出/可编程逻辑阵列输入单元8。
30	P3.1/AD1/PWM0 _L /PLAI[9]	通用输入输出端口P3.1/外部存储器接口/PWM相位0低边输出/可编程逻辑阵列输入单元9。
31	P3.2/AD2/PWM1 _H /PLAI[10]	通用输入输出端口P3.2/外部存储器接口/PWM相位1高边输出/可编程逻辑阵列输入单元10。
32	P3.3/AD3/PWM1 _L /PLAI[11]	通用输入输出端口P3.3/外部存储器接口/PWM相位1低边输出/可编程逻辑阵列输入单元11。
33	P2.4/PWM0 _H /MS0	通用输入输出端口P2.4/PWM相位0高边输出/外部存储器选择0。
34	P0.3/TRST/A16/ADC _{BUSY}	通用输入输出端口P0.3/JTAG测试端口输入，测试复位/ADC _{BUSY} 信号输出。
35	P2.5/PWM0 _L /MS1	通用输入输出端口P2.5/PWM相位0低边输出/外部存储器选择1。
36	P2.6/PWM1 _H /MS2	通用输入输出端口P2.6/PWM相位1高边输出/外部存储器选择2。
37	RST	复位输入，低电平有效。
38	P3.4/AD4/PWM2 _H /PLAI[12]	通用输入输出端口P3.4/外部存储器接口/PWM相位2高边输出/可编程逻辑阵列输入单元12。
39	P3.5/AD5/PWM2 _L /PLAI[13]	通用输入输出端口P3.5/外部存储器接口/PWM相位2低边输出/可编程逻辑阵列输入单元13。
40	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/通用输入输出端口P0.4/PWM触发外部输入/可编程逻辑阵列输出单元1/外部存储器选择1。
41	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/通用输入输出端口P0.5/ADC _{BUSY} 信号输出/可编程逻辑阵列输出单元2/外部存储器选择2。
42	P2.0/SPM9/PLAO[5]/CONV _{START}	串行复用端口。通用输入输出端口P2.0/UART/可编程逻辑阵列输出单元5/ADC起始转换输入信号。
43	P0.7/ECLK/XCLK/SPM8/PLAO[4]	串行复用端口。通用输入输出端口P0.7/外部时钟信号输出/内部时钟发生器电路输入/UART/可编程逻辑阵列输出单元4。
44	XCLKO	晶体振荡反相器输出。
45	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。

ADuC7019/20/21/22/24/25/26/27/28/29

引脚编号	引脚名称	描述
46	P3.6/AD6/PWM _{TRIP} /PLAI[14]	通用输入输出端口P 3.6/外部存储器接口/PWM安全关断/可编程逻辑阵列输入单元14。
47	P3.7/AD7/PWM _{SYNC} /PLAI[15]	通用输入输出端口P 3.7/外部存储器接口/PWM同步/可编程逻辑阵列输入单元15。
48	P2.7/PWM1 _L /MS3	通用输入输出端口P 2.7/PWM相位1低边输出/外部存储器选择3。
49	P2.1/ $\overline{\text{WS}}$ /PWM0 _H /PLAO[6]	通用输入输出端口P 2.1/外部存储器写选通/PWM相位0高边输出/可编程逻辑阵列输出单元6。
50	P2.2/ $\overline{\text{RS}}$ /PWM0 _L /PLAO[7]	通用输入输出端口P 2.2/外部存储器读选通/PWM相位0低边输出/可编程逻辑阵列输出单元7。
51	P1.7/SPM7/PLAO[0]	串行复用端口。通用输入输出端口P 1.7/UART, SPI/可编程逻辑阵列输出单元0。
52	P1.6/SPM6/PLAI[6]	串行复用端口。通用输入输出端口P 1.6/UART, SPI/可编程逻辑阵列输入单元6。
53	IOGND	GPIO地(参见表78)。通常连接到DGND。
54	IOV _{DD}	3.3 V电源, 用于GPIO(参见表78)和片内稳压器输入。
55	P4.0/AD8/PLAO[8]	通用输入输出端口P 4.0/外部存储器接口/可编程逻辑阵列输出单元8。
56	P4.1/AD9/PLAO[9]	通用输入输出端口P 4.1/外部存储器接口/可编程逻辑阵列输出单元9。
57	P1.5/SPM5/PLAI[5]/IRQ3	串行复用端口。通用输入输出端口P 1.5/UART, SPI/可编程逻辑阵列输入单元5/外部中断请求3, 高电平有效。
58	P1.4/SPM4/PLAI[4]/IRQ2	串行复用端口。通用输入输出端口P 1.4/UART, SPI/可编程逻辑阵列输入单元4/外部中断请求2, 高电平有效。
59	P1.3/SPM3/PLAI[3]	串行复用端口。通用输入输出端口P 1.3/UART, I2C1/可编程逻辑阵列输入单元3。
60	P1.2/SPM2/PLAI[2]	串行复用端口。通用输入输出端口P 1.2/UART, I2C1/可编程逻辑阵列输入单元2。
61	P1.1/SPM1/PLAI[1]	串行复用端口。通用输入输出端口P 1.1/UART, I2C0/可编程逻辑阵列输入单元1。
62	P1.0/T1/SPM0/PLAI[0]	串行复用端口。通用输入输出端口P 1.0/定时器1输入/UART, I2C0/可编程逻辑阵列输入单元0。
63	P4.2/AD10/PLAO[10]	通用输入输出端口P 4.2/外部存储器接口/可编程逻辑阵列输出单元10。
64	P4.3/AD11/PLAO[11]	通用输入输出端口P 4.3/外部存储器接口/可编程逻辑阵列输出单元11。
65	P4.4/AD12/PLAO[12]	通用输入输出端口P 4.4/外部存储器接口/可编程逻辑阵列输出单元12。
66	P4.5/AD13/PLAO[13]	通用输入输出端口P 4.5/外部存储器接口/可编程逻辑阵列输出单元13。
67	REFGND	基准地。通常连接到AGND。
68	V _{REF}	2.5 V内部基准电压。在使用内部基准电压源时必须连接至一个0.47 μF 电容。
69	DAC _{REF}	DAC外部基准电压。范围: DACGND至DACV _{DD} 。
70	DACGND	DAC地。通常连接到AGND。
71, 72	AGND	模拟地。模拟电路的地基准点。
73, 74	AV _{DD}	3.3 V模拟电源。
75	DACV _{DD}	DAC 3.3 V电压源。必须连接到AV _{DD} 。
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/比较器正相输入。
80	ADC3/CMP1	单端或差分模拟输入3/比较器反相输入。

ADUC7028

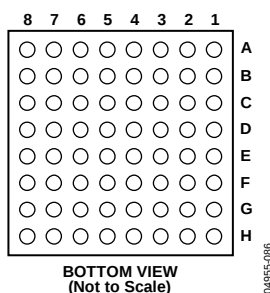


图 16. 64引脚BGA封装ADuC7028引脚配置

表14. ADuC7028引脚功能描述

引脚编号	引脚名称	描述
A1	ADC3/CMP1	单端或差分模拟输入3/比较器反相输入。
A2	DACV _{DD}	DAC 3.3 V电压源。必须连接到AV _{DD} 。
A3	AV _{DD}	3.3 V模拟电源。
A4	AGND	模拟地。模拟电路的地基准点。
A5	DACGND	DAC地。通常连接到AGND。
A6	P4.2/PLAO[10]	通用输入输出端口P4.2/可编程逻辑阵列输出单元10。
A7	P1.1/SPM1/PLAI[1]	串行复用端口。通用输入输出端口P 1.1/UART, I2C0/可编程逻辑阵列输入单元1。
A8	P1.2/SPM2/PLAI[2]	串行复用端口。通用输入输出端口P 1.2/UART, I2C1/可编程逻辑阵列输入单元2。
B1	ADC4	单端或差分模拟输入4。
B2	ADC2/CMP0	单端或差分模拟输入2/比较器正相输入。
B3	ADC1	单端或差分模拟输入1。
B4	DAC _{REF}	DAC外部基准电压。范围：DACGND至DACV _{DD} 。
B5	V _{REF}	2.5 V内部基准电压。在使用内部基准电压源时必须连接至一个0.47 μF电容。
B6	P1.0/T1/SPM0/PLAI[0]	串行复用端口。通用输入输出端口P 1.0/定时器1输入/UART, I2C0/可编程逻辑阵列输入单元0。
B7	P1.4/SPM4/PLAI[4]/IRQ2	串行复用端口。通用输入输出端口P 1.4/UART, SPI/可编程逻辑阵列输入单元4/外部中断请求2, 高电平有效。
B8	P1.3/SPM3/PLAI[3]	串行复用端口。通用输入输出端口P 1.3/UART, I2C1/可编程逻辑阵列输入单元3。
C1	ADC6	单端或差分模拟输入6。
C2	ADC5	单端或差分模拟输入5。
C3	ADC0	单端或差分模拟输入0。
C4	P4.5/PLAO[13]	通用输入输出端口P4.5/可编程逻辑阵列输出单元13。
C5	P4.3/PLAO[11]	通用输入输出端口P4.3/可编程逻辑阵列输出单元11。
C6	P4.0/PLAO[8]	通用输入输出端口P4.0/可编程逻辑阵列输出单元8。
C7	P4.1/PLAO[9]	通用输入输出端口P4.1/可编程逻辑阵列输出单元9。
C8	IOGND	GPIO地(参见表78)。通常连接到DGND。
D1	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。
D2	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。
D3	ADC7	单端或差分模拟输入7。
D4	P4.4/PLAO[12]	通用输入输出端口P4.4/可编程逻辑阵列输出单元12。
D5	P3.6/PWM _{TRIP} /PLAI[14]	通用输入输出端口P 3.6/PWM安全关断/可编程逻辑阵列输入单元14。
D6	P1.7/SPM7/PLAO[0]	串行复用端口。通用输入输出端口P 1.7/UART, SPI/可编程逻辑阵列输出单元0。

ADuC7019/20/21/22/24/25/26/27/28/29

引脚编号	引脚名称	描述
D7	P1.6/SPM6/PLAI[6]	串行复用端口。通用输入输出端口P 1.6/UART, SPI/可编程逻辑阵列输入单元6。
D8	IOV _{DD}	3.3 V电源, 用于GPIO(参见表78)和片内稳压器输入。
E1	DAC3	DAC3电压输出。
E2	DAC2	DAC2电压输出。
E3	DAC1	DAC1电压输出。
E4	P3.0/PWM0 _H /PLAI[8]	通用输入输出端口P 3.0/PWM相位0高边输出/可编程逻辑阵列输入单元8。
E5	P3.2/PWM1 _H /PLAI[10]	通用输入输出端口P 3.2/PWM相位1高边输出/可编程逻辑阵列输入单元10。
E6	P1.5/SPM5/PLAI[5]/IRQ3	串行复用端口。通用输入输出端口P 1.5/UART, SPI/可编程逻辑阵列输入单元5/外部中断请求3, 高电平有效。
E7	P3.7/PWM _{SYNC} /PLAI[15]	通用输入输出端口P 3.7/PWM同步/可编程逻辑阵列输入单元15。
E8	XCLKI	晶体振荡反相器输入和内部时钟发生器电路输入。
F1	P4.6/PLAO[14]	通用输入输出端口P4.6/可编程逻辑阵列输出单元14。
F2	TDI	JTAG测试端口输入, 测试数据输入。调试和下载访问。
F3	DAC0s	DAC0电压输出。
F4	P3.1/PWM0 _L /PLAI[9]	通用输入输出端口P 3.1/PWM相位0低边输出/可编程逻辑阵列输入单元9。
F5	P3.3/PWM1 _L /PLAI[11]	通用输入输出端口P 3.3/PWM相位1低边输出/可编程逻辑阵列输入单元11。
F6	$\overline{\text{RST}}$	复位输入, 低电平有效。
F7	P0.7/ECLK/XCLK/SPM8/PLAO[4]	串行复用端口。通用输入输出端口P0.7/外部时钟信号输出/内部时钟发生器电路输入/UART/可编程逻辑阵列输出单元4。
F8	XCLKO	晶体振荡反相器输出。
G1	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ADuC7028进入UART下载模式; 如果BM通过一个1 k Ω 电阻上拉至高电平, 则执行代码)/通用输入输出端口P0.0/电压比较器输出/可编程逻辑阵列输入单元7。
G2	P4.7/PLAO[15]	通用输入输出端口P4.7/可编程逻辑阵列输出单元15。
G3	TMS	JTAG测试端口输入, 测试模式选择。调试和下载访问。
G4	TDO	JTAG测试端口输出, 测试数据输出。调试和下载访问。
G5	P0.3/TRST/ADC _{BUSY}	通用输入输出端口P 0.3/JTAG测试端口输入, 测试复位/ADC _{BUSY} 信号输出。
G6	P3.4/PWM2 _H /PLAI[12]	通用输入输出端口P 3.4/PWM相位2高边输出/可编程逻辑阵列输入单元12。
G7	P3.5/PWM2 _L /PLAI[13]	通用输入输出端口P 3.5/PWM相位2低边输出/可编程逻辑阵列输入单元13。
G8	P2.0/SPM9/PLAO[5]/ $\overline{\text{CONV}}_{\text{START}}$	串行复用端口。通用输入输出端口P 2.0/UART/可编程逻辑阵列输出单元5/ADC起始转换输入信号。
H1	P0.6/T1/MRST/PLAO[3]	多功能引脚, 复位后输出低电平。通用输入输出端口P 0.6/定时器1输入/上电复位输出/可编程逻辑阵列输出单元3。
H2	TCK	JTAG测试端口输入, 测试时钟。调试和下载访问。
H3	IOGND	GPIO地(参见表78)。通常连接到DGND。
H4	IOV _{DD}	3.3 V电源, 用于GPIO(参见表78)和片内稳压器输入。
H5	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。
H6	DGND	内核逻辑地。
H7	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/通用输入输出端口P 0.4/PWM触发外部输入/可编程逻辑阵列输出单元1。
H8	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/通用输入输出端口P0.5/ADC _{BUSY} 信号输出/可编程逻辑阵列输出单元2。

ADUC7029

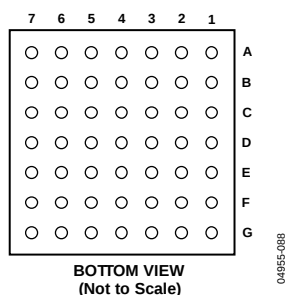


图17. 49引脚BGA封装ADuC7029引脚配置

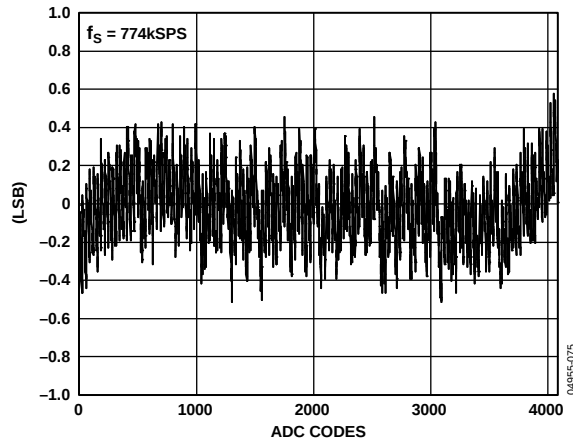
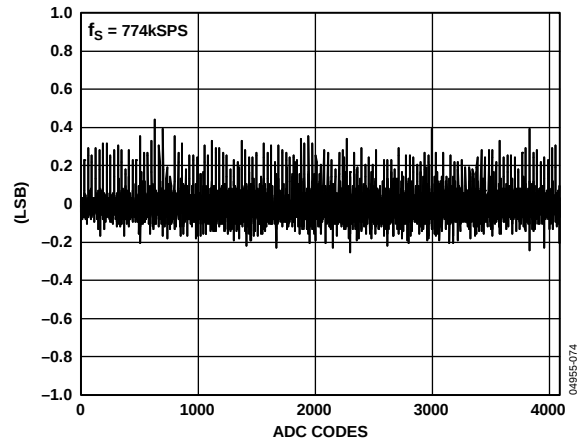
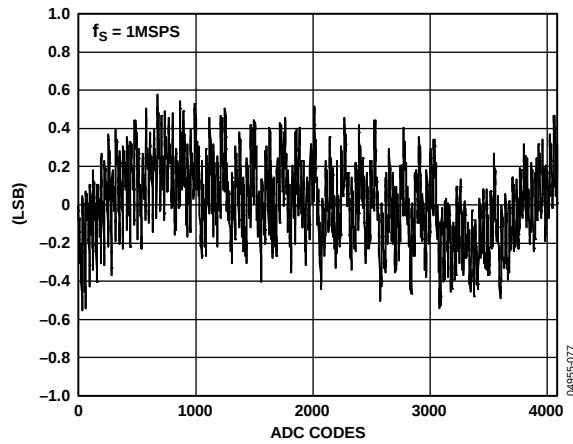
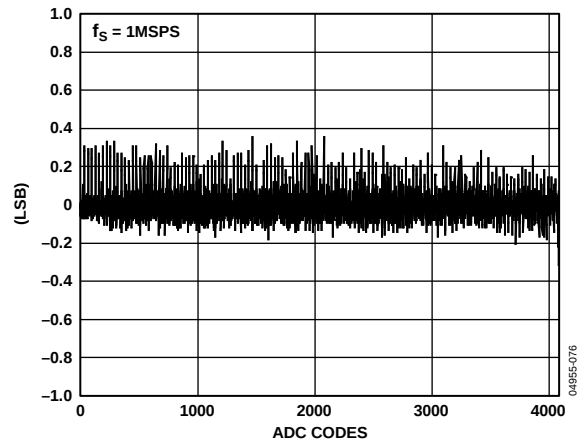
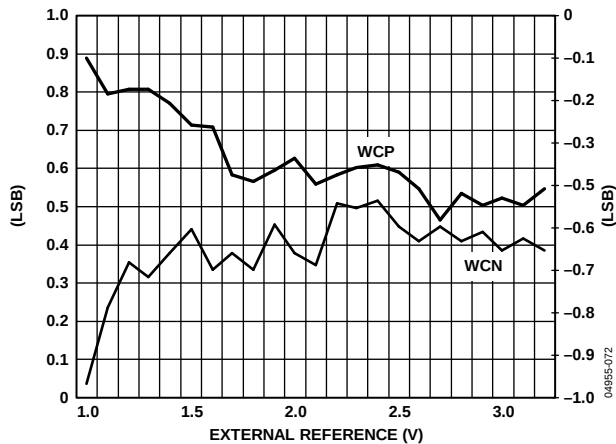
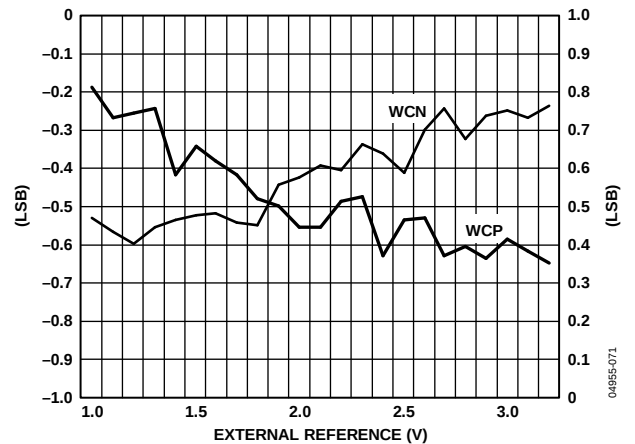
表15. ADuC7029引脚功能描述

引脚编号	引脚名称	描述
A1	ADC3/CMP1	单端或差分模拟输入3/比较器反相输入。
A2	ADC1	单端或差分模拟输入1。
A3	ADC0	单端或差分模拟输入0。
A4	AV_{DD}	3.3 V模拟电源。
A5	V_{REF}	2.5 V内部基准电压。 在使用内部基准电压源时必须连接至一个0.47 μ F电容。
A6	P1.0/T1/SPM0/PLAI[0]	串行复用端口。通用输入输出端口P 1.0/定时器1输入/UART, I2C0/ 可编程逻辑阵列输入单元0。
A7	P1.1/SPM1/PLAI[1]	串行复用端口。通用输入输出端口P 1.1/UART, I2C0/ 可编程逻辑阵列输入单元1。
B1	ADC6	单端或差分模拟输入6。
B2	ADC5	单端或差分模拟输入5。
B3	ADC4	单端或差分模拟输入4。
B4	AGND	模拟地。模拟电路的地基准点。
B5	DAC_{REF}	DAC外部基准电压。范围： DAC_{GND} 至 $DACV_{DD}$ 。
B6	P1.4/SPM4/PLAI[4]/IRQ2	串行复用端口。通用输入输出端口P 1.4/UART, SPI/可编程逻辑阵列输入单元4/ 外部中断请求2, 高电平有效。
B7	P1.3/SPM3/PLAI[3]	串行复用端口。通用输入输出端口P 1.3/UART, I2C1/ 可编程逻辑阵列输入单元3。
C1	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。
C2	AGND	模拟地。模拟电路的地基准点。
C3	ADC2/CMP0	单端或差分模拟输入2/比较器正相输入。
C4	IOGND	GPIO地(参见表78)。通常连接到DGND。
C5	P1.2/SPM2/PLAI[2]	串行复用端口。通用输入输出端口P 1.2/UART, I2C1/ 可编程逻辑阵列输入单元2。
C6	P1.6/SPM6/PLAI[6]	串行复用端口。通用输入输出端口P 1.6/UART, SPI/ 可编程逻辑阵列输入单元6。
C7	P1.5/SPM5/PLAI[5]/IRQ3	串行复用端口。通用输入输出端口P 1.5/UART, SPI/ 可编程逻辑阵列输入单元5/外部中断请求3, 高电平有效。
D1	DAC0	DAC0电压输出。
D2	DAC3	DAC3电压输出。
D3	DAC1	DAC1电压输出。
D4	P3.3/PWM1 _L /PLAI[11]	通用输入输出端口P 3.3/PWM相位1低边输出/ 可编程逻辑阵列输入单元11。
D5	P3.4/PWM2 _H /PLAI[12]	通用输入输出端口P 3.4/PWM相位2高边输出/ 可编程逻辑阵列输入单元12。
D6	P3.6/PWM _{TRIP} /PLAI[14]	通用输入输出端口P 3.6/PWM安全关断/ 可编程逻辑阵列输入单元14。
D7	P1.7/SPM7/PLAO[0]	串行复用端口。通用输入输出端口P 1.7/UART, SPI/ 可编程逻辑阵列输出单元0。

ADuC7019/20/21/22/24/25/26/27/28/29

引脚编号	引脚名称	描述
E1	TMS	JTAG测试端口输入，测试模式选择。调试和下载访问。
E2	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ADuC7029进入UART下载模式；如果BM通过一个1 kΩ电阻上拉至高电平，则执行代码)/通用输入输出端口P0.0/电压比较器输出/可编程逻辑阵列输入单元7。
E3	DAC2	DAC2电压输出。
E4	IOV _{DD}	3.3 V电源，用于GPIO(参见表78)和片内稳压器输入。
E5	P3.2/PWM1 _H /PLAI[10]	通用输入输出端口P 3.2/PWM相位1高边输出/可编程逻辑阵列输入单元10。
E6	P3.5/PWM2 _L /PLAI[13]	通用输入输出端口P 3.5/PWM相位2低边输出/可编程逻辑阵列输入单元13。
E7	P0.7/ECLK/XCLK/SPM8/PLAO[4]	串行复用端口。通用输入输出端口P0.7/外部时钟信号输出/内部时钟发生器电路输入/UART/可编程逻辑阵列输出单元4。
F1	TDI	JTAG测试端口输入，测试数据输入。调试和下载访问。
F2	P0.6/T1/MRST/PLAO[3]	多功能引脚，复位后输出低电平。通用输入输出端口P 0.6/定时器1输入/上电复位输出/可编程逻辑阵列输出单元3。
F3	IOGND	GPIO地(参见表78)。通常连接到DGND。
F4	P3.1/PWM0 _L /PLAI[9]	通用输入输出端口P 3.1/PWM相位0低边输出/可编程逻辑阵列输入单元9。
F5	P3.0/PWM0 _H /PLAI[8]	通用输入输出端口P 3.0/PWM相位0高边输出/可编程逻辑阵列输入单元8。
F6	$\overline{\text{RST}}$	复位输入，低电平有效。
F7	P2.0/SPM9/PLAO[5]/ $\overline{\text{CONV}}_{\text{START}}$	串行复用端口。通用输入输出端口P 2.0/UART/可编程逻辑阵列输出单元5/ADC起始转换输入信号。
G1	TCK	JTAG测试端口输入，测试时钟。调试和下载访问。
G2	TDO	JTAG测试端口输出，测试数据输出。调试和下载访问。
G3	LV _{DD}	片内稳压器2.6 V输出。该输出只能通过一个0.47 μF电容器连接至DGND。
G4	DGND	内核逻辑地。
G5	P0.3/TRST/ADC _{BUSY}	通用输入输出端口P 0.3/JTAG测试端口输入，测试复位/ADC _{BUSY} 信号输出。
G6	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/通用输入输出端口P 0.4/PWM触发外部输入/可编程逻辑阵列输出单元1。
G7	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/通用输入输出端口P0.5/ADC _{BUSY} 信号输出/可编程逻辑阵列输出单元2。

典型性能参数

图18. $f_s = 774$ kSPS时典型INL误差图21. $f_s = 774$ kSPS时典型DNL误差图19. $f_s = 1$ MSPS时典型INL误差图22. $f_s = 1$ MSPS时典型DNL误差图20. $f_s = 774$ kSPS时典型最差(正(WCP)和负(WCN))
INL误差与 V_{REF} 关系图23. $f_s = 774$ kSPS时典型最差(正(WCP)和负(WCN))
DNL误差与 V_{REF} 关系

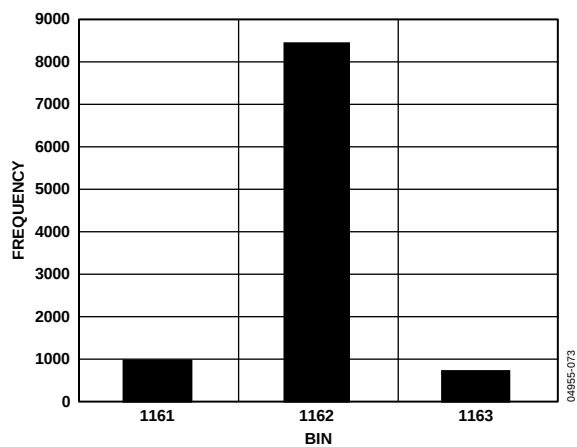


图24. $f_s = 774 \text{ kSPS}$ 、 $V_{IN} = 0.7 \text{ V}$ 时代码直方图

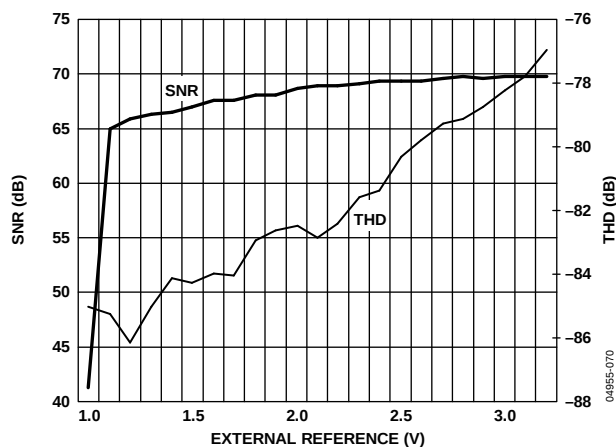


图27.典型动态性能与 V_{REF} 关系

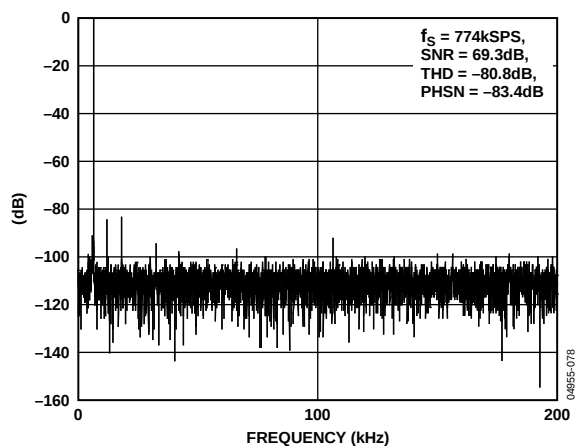


图25. $f_s = 774 \text{ kSPS}$ 时动态性能

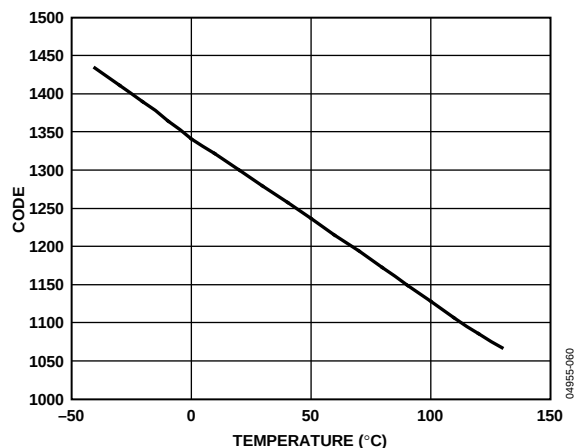


图28. 片内温度传感器电压输出与温度的关系

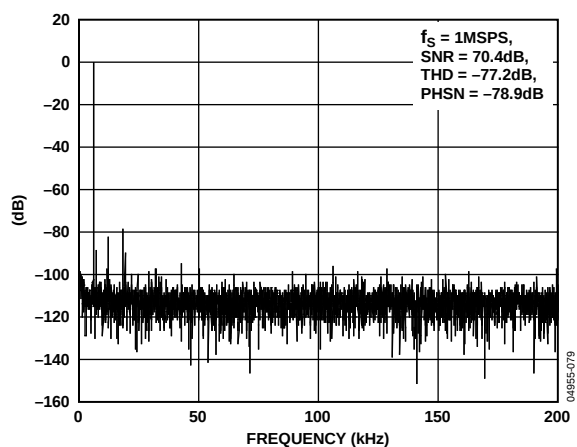


图26. $f_s = 1 \text{ MSPS}$ 时动态性能

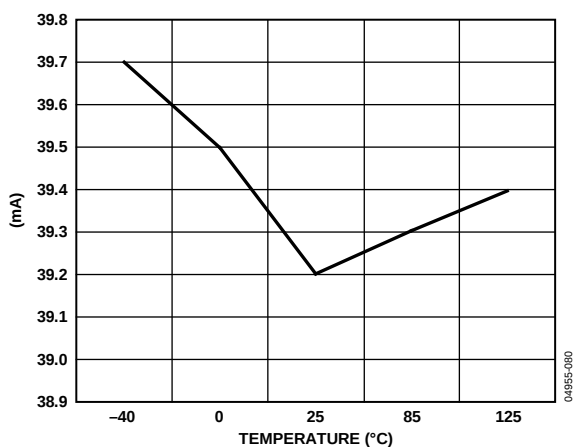


图29. $CD = 0$ 时功耗与温度的关系

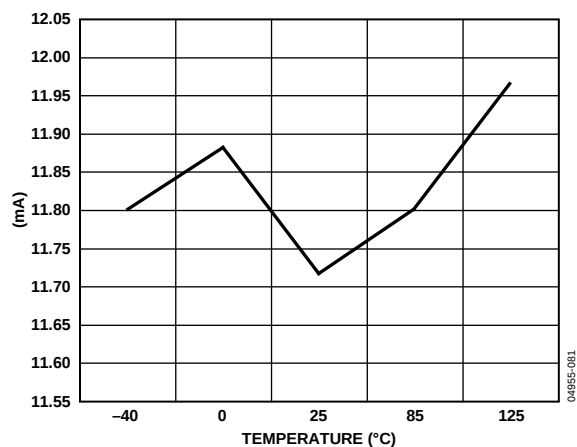


图30. CD = 3时功耗与温度的关系

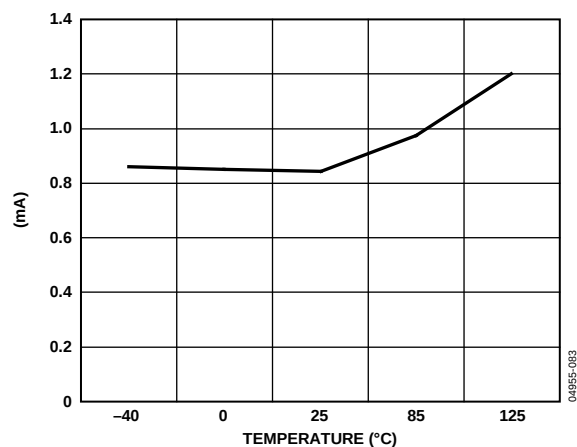


图32. 睡眠模式下功耗与温度的关系

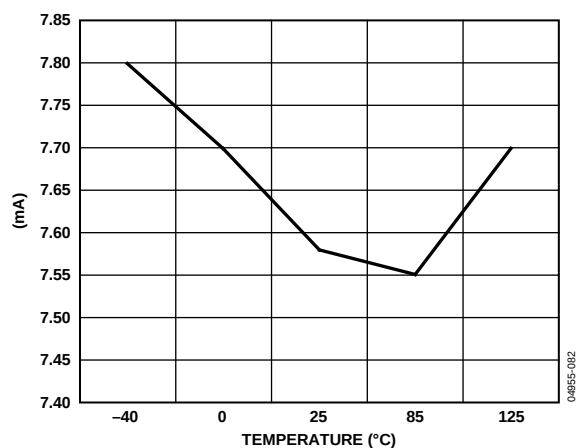


图31. CD = 7时功耗与温度的关系

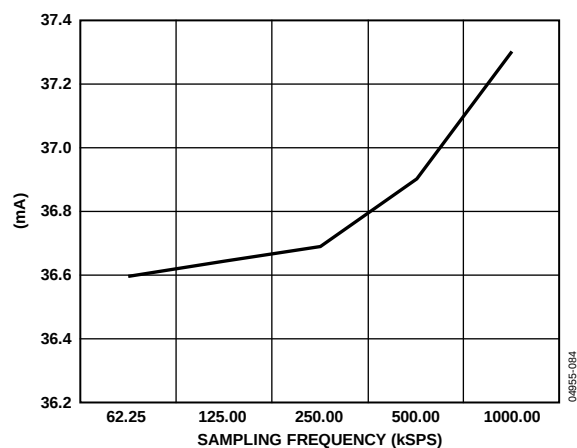


图33. 功耗与采样频率的关系

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1/2 LSB的点。

差分非线性(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+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(SINAD)

在ADC输出端所测量到的信号与(噪声+失真)的比值。这里的信号是基波幅值的均方根值。噪声为除了直流信号以外一直到半采样频率($f_s/2$)的所有非基波信号均方根和。

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

对于一个正弦波输入的理想N位转换器，信号与噪声+失真比值的理论计算值为：

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

因此，对于12位转换器，该值为74 dB。

总谐波失真(THD)

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。

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

THUMB模式(T)

一条ARM指令的长度为32位。ARM7TDMI处理器支持压缩至16位的第二指令集，即Thumb指令集。用thumb指令集替代ARM指令集，可以更为快速的从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，这种异常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)，用于调用操作系统。

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

ARM寄存器

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

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

异常发生后，异常模式专用的寄存器将取代某些标准寄存器。所有的异常模式都有各自的替换寄存器组，用于堆栈指针(R13)和链接寄存器(R14)，如图34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。

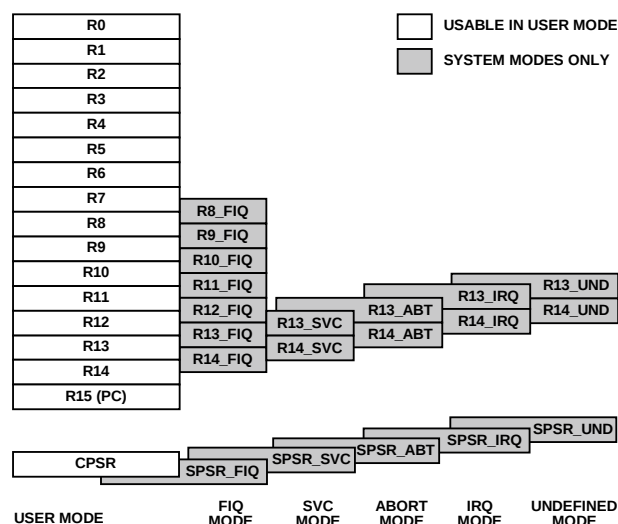


图34. 寄存器结构图

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

- 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中断矢量地址)处的指令。最长总延迟时间为50个处理器周期,在系统采用连续41.78 MHz处理器时钟时,略小于1.2 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

ADuC7019/20/21/22/24/25/26/27/28/29有两个独立存储器模块：8 kB的SRAM和64 kB的片内Flash/EE存储器。片内Flash/EE存储器有62 kB可供用户使用，剩余2 kB保留用于厂家配置的引导页面。这两个存储器模块的映射如图35所示。

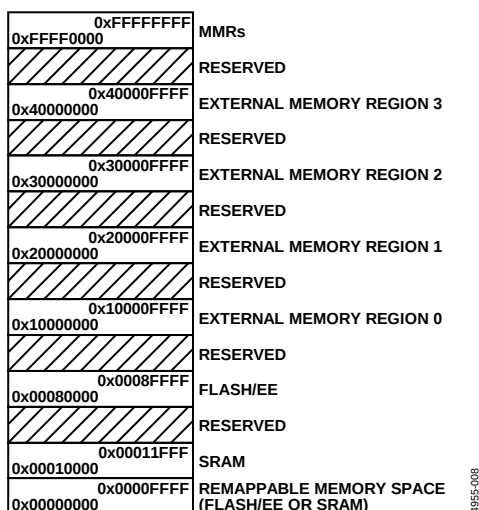


图35. 物理存储器映射图

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REMAP MMR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。

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35所示。

ADuC7019/20/21/22/24/25/26/27/28/29的存储器被配置成从小到大顺序格式，这就是说，LSB位于最低字节地址，MSB位于最高字节地址。

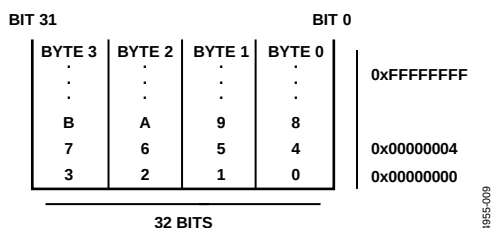


图 36. 从小到大顺序格式

FLASH/EE存储器

64 kB的Flash/EE存储器为 $32 \text{ k} \times 16$ 位格式，其中 $31 \text{ k} \times 16$ 位是用户空间， $1 \text{ k} \times 16$ 位保留用于芯片内核。Flash/EE的页面大小为512个字节。

62 KB的片内Flash/EE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模式下每一次执行一个指令(32位指令)，必须读取两次Flash/EE存储器。因此，当从Flash/EE中执行程序时，为得到更高的存取速度，建议使用thumb代码。以thumb代码模式存取Flash/EE的最大速度为41.78 MHz，而相应的以全ARM模式为20.89 MHz。更多关于Flash/EE存取时间的描述可参见SRAM和Flash/EE中的执行时间部分。

SRAM

用户可以使用8 KB的SRAM，它的组织形式为 $2 \text{ k} \times 32$ 位，即两个字。如果SRAM被配置成32位宽的存储器阵列，ARM代码可以直接在SRAM中以41.78 MHz的速度执行。更多关于SRAM存取时间的描述可参见SRAM和Flash/EE中的执行时间部分。

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37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16为所有的MMR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、GPIO(见表78)和PWM以外，ADuC7019/20/21/22/24/25/26/27/28/29的其它所有外围设备都连接到APB总线。

0xFFFFFFFF	
0xFFFFFC3C	PWM
0xFFFFFC00	
0xFFFFF820	FLASH CONTROL INTERFACE
0xFFFFF800	
0xFFFFF46C	
0xFFFFF400	GPIO
0xFFFF0B54	
0xFFFF0B00	PLA
0xFFFF0A14	
0xFFFF0A00	SPI
0xFFFF0948	
0xFFFF0900	I2C1
0xFFFF0848	
0xFFFF0800	I2C0
0xFFFF0730	
0xFFFF0700	UART
0xFFFF0620	
0xFFFF0600	DAC
0xFFFF0538	
0xFFFF0500	ADC
0xFFFF0490	
0xFFFF048C	BAND GAP REFERENCE
0xFFFF0448	
0xFFFF0440	POWER SUPPLY MONITOR
0xFFFF0420	
0xFFFF0404	PLL AND OSCILLATOR CONTROL
0xFFFF0370	
0xFFFF0360	WATCHDOG TIMER
0xFFFF0350	
0xFFFF0340	WAKE-UP TIMER
0xFFFF0334	
0xFFFF0320	GENERAL-PURPOSE TIMER
0xFFFF0310	
0xFFFF0300	TIMER 0
0xFFFF0238	
0xFFFF0220	REMAP AND SYSTEM CONTROL
0xFFFF0110	
0xFFFF0000	INTERRUPT CONTROLLER

图 37. 存储器映射寄存器

表16. 完整MMR列表

地址	地址	字节	访问类型	默认值	页码
中断请求IRQ基地址 = 0xFFFF0000					
0x0000	IRQSTA	4	R	0x00000000	78
0x0004	IRQSIG ¹	4	R	0x00XXX000	78
0x0008	IRQEN	4	R/W	0x00000000	78
0x000C	IRQCLR	4	W	0x00000000	78
0x0010	SWICFG	4	W	0x00000000	79
0x0100	FIQSTA	4	R	0x00000000	79
0x0104	FIQSIG ¹	4	R	0x00XXX000	79
0x0108	FIQEN	4	R/W	0x00000000	79
0x010C	FIQCLR	4	W	0x00000000	79

¹ 取决于外部中断引脚P0.4、P0.5、P1.4和P1.5的电平。

系统控制基地址 = 0xFFFF0200

0x0220	REMAP	1	R/W	0xXX ¹	51
0x0230	RSTSTA	1	R/W	0x01	51
0x0234	RSTCLR	1	W	0x00	51

¹ 取决于具体的器件型号。

定时器基地址 = 0xFFFF0300

0x0300	T0LD	2	R/W	0x0000	80
0x0304	T0VAL	2	R	0xFFFF	80
0x0308	T0CON	2	R/W	0x0000	80
0x030C	T0CLR	1	W	0xFF	80
0x0320	T1LD	4	R/W	0x00000000	81
0x0324	T1VAL	4	R	0xFFFFFFFF	81
0x0328	T1CON	2	R/W	0x0000	81
0x032C	T1CLR	1	W	0xFF	82
0x0330	T1CAP	4	R/W	0x00000000	82
0x0340	T2LD	4	R/W	0x00000000	82
0x0344	T2VAL	4	R	0xFFFFFFFF	82
0x0348	T2CON	2	R/W	0x0000	82
0x034C	T2CLR	1	W	0xFF	83
0x0360	T3LD	2	R/W	0x0000	83
0x0364	T3VAL	2	R	0xFFFF	83
0x0368	T3CON	2	R/W	0x0000	83
0x036C	T3CLR	1	W	0x00	84

锁相环PLL基地址 = 0xFFFF0400

0x0404	POWKEY1	2	W	0x0000	56
0x0408	POWCON	2	R/W	0x0003	56
0x040C	POWKEY2	2	W	0x0000	56
0x0410	PLLKEY1	2	W	0x0000	56
0x0414	PLLCON	1	R/W	0x21	56
0x0418	PLLKEY2	2	W	0x0000	56

PSM基地址 = 0xFFFF0440

0x0440	PSMCON	2	R/W	0x0008	53
0x0444	CMPCON	2	R/W	0x0000	54

ADuC7019/20/21/22/24/25/26/27/28/29

地址	名称	字节	访问类型	默认值	页码
基准电压基地址 = 0xFFFF0480					
0x048C	REFCON	1	R/W	0x00	46

模数转换器基地址 = 0xFFFF0500

0x0500	ADCCON	2	R/W	0x0600	42
0x0504	ADCCP	1	R/W	0x00	43
0x0508	ADCCN	1	R/W	0x01	43
0x050C	ADCSTA	1	R	0x00	44
0x0510	ADCDAT	4	R	0x00000000	44
0x0514	ADCRST	1	R/W	0x00	44
0x0530	ADCGN	2	R/W	0x0200	44
0x0534	ADCOF	2	R/W	0x0200	44

数模转换器基地址 = 0xFFFF0600

0x0600	DAC0CON	1	R/W	0x00	52
0x0604	DAC0DAT	4	R/W	0x00000000	52
0x0608	DAC1CON	1	R/W	0x00	52
0x060C	DAC1DAT	4	R/W	0x00000000	52
0x0610	DAC2CON	1	R/W	0x00	52
0x0614	DAC2DAT	4	R/W	0x00000000	52
0x0618	DAC3CON	1	R/W	0x00	52
0x061C	DAC3DAT	4	R/W	0x00000000	52

UART基地址 = 0xFFFF0700

0x0700	COMTX	1	R/W	0x00	66
	COMRX	1	R	0x00	66
	COMDIV0	1	R/W	0x00	66
0x0704	COMIEN0	1	R/W	0x00	66
	COMDIV1	1	R/W	0x00	66
0x0708	COMIID0	1	R	0x01	67
0x070C	COMCON0	1	R/W	0x00	67
0x0710	COMCON1	1	R/W	0x00	67
0x0714	COMSTA0	1	R	0x60	67
0x0718	COMSTA1	1	R	0x00	68
0x071C	COMSCR	1	R/W	0x00	68
0x0720	COMIEN1	1	R/W	0x04	68
0x0724	COMIID1	1	R	0x01	68
0x0728	COMADR	1	R/W	0xAA	69
0x072C	COMDIV2	2	R/W	0x0000	68

地址	名称	字节	访问类型	默认值	页码
I2C0基地址 = 0xFFFF0800					

0x0800	I2C0MSTA	1	R/W	0x00	71
0x0804	I2C0SSTA	1	R	0x01	71
0x0808	I2C0SRX	1	R	0x00	72
0x080C	I2C0STX	1	W	0x00	72
0x0810	I2C0MRX	1	R	0x00	72
0x0814	I2C0MTX	1	W	0x00	72
0x0818	I2C0CNT	1	R/W	0x00	72
0x081C	I2C0ADR	1	R/W	0x00	72
0x0824	I2C0BYTE	1	R/W	0x00	72
0x0828	I2C0ALT	1	R/W	0x00	73
0x082C	I2C0CFG	1	R/W	0x00	73
0x0830	I2C0DIV	2	R/W	0x1F1F	74
0x0838	I2C0ID0	1	R/W	0x00	74
0x083C	I2C0ID1	1	R/W	0x00	74
0x0840	I2C0ID2	1	R/W	0x00	74
0x0844	I2C0ID3	1	R/W	0x00	74
0x0848	I2C0CCNT	1	R/W	0x01	74
0x084C	I2C0FSTA	2	R/W	0x0000	74

I2C1基地址 = 0xFFFF0900

0x0900	I2C1MSTA	1	R/W	0x00	71
0x0904	I2C1SSTA	1	R	0x01	71
0x0908	I2C1SRX	1	R	0x00	72
0x090C	I2C1STX	1	W	0x00	72
0x0910	I2C1MRX	1	R	0x00	72
0x0914	I2C1MTX	1	W	0x00	72
0x0918	I2C1CNT	1	R/W	0x00	72
0x091C	I2C1ADR	1	R/W	0x00	72
0x0924	I2C1BYTE	1	R/W	0x00	72
0x0928	I2C1ALT	1	R/W	0x00	73
0x092C	I2C1CFG	1	R/W	0x00	73
0x0930	I2C1DIV	2	R/W	0x1F1F	74
0x0938	I2C1ID0	1	R/W	0x00	74
0x093C	I2C1ID1	1	R/W	0x00	74
0x0940	I2C1ID2	1	R/W	0x00	74
0x0944	I2C1ID3	1	R/W	0x00	74
0x0948	I2C1CCNT	1	R/W	0x01	74
0x094C	I2C1FSTA	2	R/W	0x0000	74

SPI基地址 = 0xFFFF0A00

0x0A00	SPISTA	1	R	0x00	70
0x0A04	SPIRX	1	R	0x00	70
0x0A08	SPI TX	1	W	0x00	70
0x0A0C	SPIDIV	1	R/W	0x1B	70
0x0A10	SPICON	2	R/W	0x0000	70

ADuC7019/20/21/22/24/25/26/27/28/29

地址	名称	字节	访问类型	默认值	页码
PLA基地址 = 0xFFFF0B00					
0x0B00	PLAELM0	2	R/W	0x0000	75
0x0B04	PLAELM1	2	R/W	0x0000	75
0x0B08	PLAELM2	2	R/W	0x0000	75
0x0B0C	PLAELM3	2	R/W	0x0000	75
0x0B10	PLAELM4	2	R/W	0x0000	75
0x0B14	PLAELM5	2	R/W	0x0000	75
0x0B18	PLAELM6	2	R/W	0x0000	75
0x0B1C	PLAELM7	2	R/W	0x0000	75
0x0B20	PLAELM8	2	R/W	0x0000	75
0x0B24	PLAELM9	2	R/W	0x0000	75
0x0B28	PLAELM10	2	R/W	0x0000	75
0x0B2C	PLAELM11	2	R/W	0x0000	75
0x0B30	PLAELM12	2	R/W	0x0000	75
0x0B34	PLAELM13	2	R/W	0x0000	75
0x0B38	PLAELM14	2	R/W	0x0000	75
0x0B3C	PLAELM15	2	R/W	0x0000	75
0x0B40	PLAELM16	2	R/W	0x0000	75
0x0B44	PLAELM17	2	R/W	0x0000	75
0x0B48	PLAELM18	2	R/W	0x0000	75
0x0B4C	PLAELM19	2	R/W	0x0000	75
0x0B50	PLAELM20	2	R/W	0x0000	75
0x0B54	PLAELM21	2	R/W	0x0000	75
0x0B58	PLAELM22	2	R/W	0x0000	75
0x0B5C	PLAELM23	2	R/W	0x0000	75
0x0B60	PLAELM24	2	R/W	0x0000	75
0x0B64	PLAELM25	2	R/W	0x0000	75
0x0B68	PLAELM26	2	R/W	0x0000	75
0x0B6C	PLAELM27	2	R/W	0x0000	75
0x0B70	PLAELM28	2	R/W	0x0000	75
0x0B74	PLAELM29	2	R/W	0x0000	75
0x0B78	PLAELM30	2	R/W	0x0000	75
0x0B7C	PLAELM31	2	R/W	0x0000	75
0x0B80	PLAELM32	2	R/W	0x0000	75
0x0B84	PLAELM33	2	R/W	0x0000	75
0x0B88	PLAELM34	2	R/W	0x0000	75
0x0B8C	PLAELM35	2	R/W	0x0000	75
0x0B90	PLAELM36	2	R/W	0x0000	75
0x0B94	PLAELM37	2	R/W	0x0000	75
0x0B98	PLAELM38	2	R/W	0x0000	75
0x0B9C	PLAELM39	2	R/W	0x0000	75
0x0BA0	PLAELM40	2	R/W	0x0000	75
0x0BA4	PLAELM41	2	R/W	0x0000	75
0x0BA8	PLAELM42	2	R/W	0x0000	75
0x0BAC	PLAELM43	2	R/W	0x0000	75
0x0BB0	PLAELM44	2	R/W	0x0000	75
0x0BB4	PLAELM45	2	R/W	0x0000	75
0x0BB8	PLAELM46	2	R/W	0x0000	75
0x0BBC	PLAELM47	2	R/W	0x0000	75
0x0BD0	PLAELM48	2	R/W	0x0000	75
0x0BD4	PLAELM49	2	R/W	0x0000	75
0x0BD8	PLAELM50	2	R/W	0x0000	75
0x0BDC	PLAELM51	2	R/W	0x0000	75
0x0BE0	PLAELM52	2	R/W	0x0000	75
0x0BE4	PLAELM53	2	R/W	0x0000	75
0x0BE8	PLAELM54	2	R/W	0x0000	75
0x0BEC	PLAELM55	2	R/W	0x0000	75
0x0BF0	PLAELM56	2	R/W	0x0000	75
0x0BF4	PLAELM57	2	R/W	0x0000	75
0x0BF8	PLAELM58	2	R/W	0x0000	75
0x0BFC	PLAELM59	2	R/W	0x0000	75
0x0C00	PLAELM60	2	R/W	0x0000	75
0x0C04	PLAELM61	2	R/W	0x0000	75
0x0C08	PLAELM62	2	R/W	0x0000	75
0x0C0C	PLAELM63	2	R/W	0x0000	75
0x0C10	PLAELM64	2	R/W	0x0000	75
0x0C14	PLAELM65	2	R/W	0x0000	75
0x0C18	PLAELM66	2	R/W	0x0000	75
0x0C1C	PLAELM67	2	R/W	0x0000	75
0x0C20	PLAELM68	2	R/W	0x0000	75
0x0C24	PLAELM69	2	R/W	0x0000	75
0x0C28	PLAELM70	2	R/W	0x0000	75
0x0C2C	PLAELM71	2	R/W	0x0000	75
0x0C30	PLAELM72	2	R/W	0x0000	75
0x0C34	PLAELM73	2	R/W	0x0000	75
0x0C38	PLAELM74	2	R/W	0x0000	75
0x0C3C	PLAELM75	2	R/W	0x0000	75
0x0C40	PLAELM76	2	R/W	0x0000	75
0x0C44	PLAELM77	2	R/W	0x0000	75
0x0C48	PLAELM78	2	R/W	0x0000	75
0x0C4C	PLAELM79	2	R/W	0x0000	75
0x0C50	PLAELM80	2	R/W	0x0000	75
0x0C54	PLAELM81	2	R/W	0x0000	75
0x0C58	PLAELM82	2	R/W	0x0000	75
0x0C5C	PLAELM83	2	R/W	0x0000	75
0x0C60	PLAELM84	2	R/W	0x0000	75
0x0C64	PLAELM85	2	R/W	0x0000	75
0x0C68	PLAELM86	2	R/W	0x0000	75
0x0C6C	PLAELM87	2	R/W	0x0000	75
0x0C70	PLAELM88	2	R/W	0x0000	75
0x0C74	PLAELM89	2	R/W	0x0000	75
0x0C78	PLAELM90	2	R/W	0x0000	75
0x0C7C	PLAELM91	2	R/W	0x0000	75
0x0C80	PLAELM92	2	R/W	0x0000	75
0x0C84	PLAELM93	2	R/W	0x0000	75
0x0C88	PLAELM94	2	R/W	0x0000	75
0x0C8C	PLAELM95	2	R/W	0x0000	75
0x0C90	PLAELM96	2	R/W	0x0000	75
0x0C94	PLAELM97	2	R/W	0x0000	75
0x0C98	PLAELM98	2	R/W	0x0000	75
0x0C9C	PLAELM99	2	R/W	0x0000	75
0x0CA0	PLAELM100	2	R/W	0x0000	75
0x0CA4	PLAELM101	2	R/W	0x0000	75
0x0CA8	PLAELM102	2	R/W	0x0000	75
0x0CAC	PLAELM103	2	R/W	0x0000	75
0x0CB0	PLAELM104	2	R/W	0x0000	75
0x0CB4	PLAELM105	2	R/W	0x0000	75
0x0CB8	PLAELM106	2	R/W	0x0000	75
0x0CBC	PLAELM107	2	R/W	0x0000	75
0x0CC0	PLAELM108	2	R/W	0x0000	75
0x0CC4	PLAELM109	2	R/W	0x0000	75
0x0CC8	PLAELM110	2	R/W	0x0000	75
0x0CCC	PLAELM111	2	R/W	0x0000	75
0x0CD0	PLAELM112	2	R/W	0x0000	75
0x0CD4	PLAELM113	2	R/W	0x0000	75
0x0CD8	PLAELM114	2	R/W	0x0000	75
0x0CDC	PLAELM115	2	R/W	0x0000	75
0x0CE0	PLAELM116	2	R/W	0x0000	75
0x0CE4	PLAELM117	2	R/W	0x0000	75
0x0CE8	PLAELM118	2	R/W	0x0000	75
0x0CEC	PLAELM119	2	R/W	0x0000	75
0x0CF0	PLAELM120	2	R/W	0x0000	75
0x0CF4	PLAELM121	2	R/W	0x0000	75
0x0CF8	PLAELM122	2	R/W	0x0000	75
0x0CFC	PLAELM123	2	R/W	0x0000	75
0x0D00	PLAELM124	2	R/W	0x0000	75
0x0D04	PLAELM125	2	R/W	0x0000	75
0x0D08	PLAELM126	2	R/W	0x0000	75
0x0D0C	PLAELM127	2	R/W	0x0000	75
0x0D10	PLAELM128	2	R/W	0x0000	75
0x0D14	PLAELM129	2	R/W	0x0000	75
0x0D18	PLAELM130	2	R/W	0x0000	75
0x0D1C	PLAELM131	2	R/W	0x0000	75
0x0D20	PLAELM132	2	R/W	0x0000	75
0x0D24	PLAELM133	2	R/W	0x0000	75
0x0D28	PLAELM134	2	R/W	0x0000	75
0x0D2C	PLAELM135	2	R/W	0x0000	75
0x0D30	PLAELM136	2	R/W	0x0000	75
0x0D34	PLAELM137	2	R/W	0x0000	75
0x0D38	PLAELM138	2	R/W	0x0000	75
0x0D3C	PLAELM139	2	R/W	0x0000	75
0x0D40	PLAELM140	2	R/W	0x0000	75
0x0D44	PLAELM141	2	R/W	0x0000	75
0x0D48	PLAELM142	2	R/W	0x0000	75
0x0D4C	PLAELM143	2	R/W	0x0000	75
0x0D50	PLAELM144	2	R/W	0x0000	75
0x0D54	PLAELM145	2	R/W	0x0000	75
0x0D58	PLAELM146	2	R/W	0x0000	75
0x0D5C	PLAELM147	2	R/W	0x0000	75
0x0D60	PLAELM148	2	R/W	0x0000	75
0x0D64	PLAELM149	2	R/W	0x0000	75
0x0D68	PLAELM150	2	R/W	0x0000	75
0x0D6C	PLAELM151	2	R/W	0x0000	75
0x0D70	PLAELM152	2	R/W	0x0000	75
0x0D74	PLAELM153	2	R/W	0x0000	75
0x0D78	PLAELM154	2	R/W	0x0000	75
0x0D7C	PLAELM155	2	R/W	0x0000	75
0x0D80	PLAELM156	2	R/W	0x0000	75
0x0D84	PLAELM157	2	R/W	0x0000	75
0x0D88	PLAELM158	2	R/W	0x0000	75
0x0D8C	PLAELM159	2	R/W	0x0000	75
0x0D90	PLAELM160	2	R/W	0x0000	75
0x0D94	PLAELM161	2	R/W	0x0000	75
0x0D98	PLAELM162	2	R/W	0x0000	75
0x0D9C	PLAELM163	2	R/W	0x0000	75
0x0DA0	PLAELM164	2	R/W	0x0000	75
0x0DA4	PLAELM165	2	R/W	0x0000	75
0x0DA8	PLAELM166	2	R/W	0x0000	75
0x0DAC	PLAELM167	2	R/W	0x0000	75
0x0DB0	PLAELM168	2	R/W	0x0000	75
0x0DB4	PLAELM169	2	R/W	0x0000	75
0x0DB8	PLAELM170	2	R/W	0x0000	75
0x0DBC	PLAELM171	2	R/W	0x0000	75
0x0DC0	PLAELM172	2	R/W	0x0000	75
0x0DC4	PLAELM173	2	R/W	0x0000	75
0x0DC8	PLAELM174	2	R/W	0x0000	75
0x0DCC	PLAELM175	2	R/W	0x0000	75
0x0DD0	PLAELM176	2	R/W	0x0000	75
0x0DD4	PLAELM177	2	R/W	0x0000	75
0x0DD8	PLAELM178	2	R/W	0x0000	75
0x0DDC	PLAELM179	2	R/W	0x0000	75
0x0DE0	PLAELM180	2	R/W	0x0000	75
0x0DE4	PLAELM181	2	R/W	0x0000	75
0x0DE8	PLAELM182	2	R/W	0x0000	75
0x0DEC	PLAELM183	2	R/W	0x0000	75
0x0DF0	PLAELM184	2	R/W	0x0000	75
0x0DF4	PLAELM185	2	R/W	0x0000	75
0x0DF8	PLAELM186	2	R/W	0x0000	75
0x0DFC	PLAELM187	2	R/W	0x0000	75
0x0E00	PLAELM188	2	R/W	0x0000	75
0x0E04	PLAELM189	2	R/W	0x0000	75
0x0E08	PLAELM190	2	R/W	0x0000	75
0x0E0C	PLAELM191	2	R/W	0x0000	75
0x0E10	PLAELM192	2	R/W	0x0000	75
0x0E14	PLAELM193	2	R/W	0x0000	75
0x0E18	PLAELM194	2	R/W	0x0000	75
0x0E1C	PLAELM195	2	R/W	0x0000	75
0x0E20	PLAELM196	2	R/W	0x0000	75
0x0E24	PLAELM197	2	R/W	0x0000	75
0x0E28	PLAELM198	2	R/W	0x0000	75
0x0E2C	PLAELM199	2	R/W	0x0000	75
0x0E30	PLAELM200	2	R/W	0x0000	75
0x0E34	PLAELM201	2	R/W	0x0000	75
0x0E38	PLAELM202	2	R/W	0x0000	75
0x0E3C	PLAELM203	2	R/W	0x0000	75
0x0E40	PLAELM204	2	R/W	0x0000	75
0x0E44	PLAELM205	2	R/W	0x0000	75
0x0E48	PLAELM206	2	R/W	0x0000	75
0x0E4C	PLAELM207	2	R/W	0x0000	75
0x0E50	PLAELM208	2	R/W	0x0000	75
0x0E54	PLAELM209	2	R/W	0x0000	75
0x0E58	PLAELM210	2	R/W	0x0000	75
0x0E5C	PLAELM211	2	R/W	0x0000	75
0x0E60	PLAELM212	2	R/W	0x0000	75

ADC电路概览

该模数转换器(ADC)集成一个快速多通道12位ADC, 工作电压为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。根据输入信号配置, 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 $2 V_{REF}$ (见图38)。

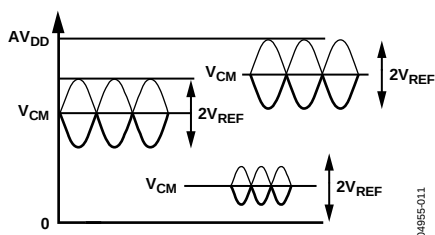


图38. 全差分模式下的平衡信号示例

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

单次或连续转换模式可由软件启动。外部 $\overline{CONV_START}$ 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, 这样就可以很方便的形成一个内部温度传感器通道, 用于测量芯片温度, 测量精度典型值为 $\pm 3^\circ\text{C}$ 。

传递函数

伪差分模式和单端模式

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

$$1 \text{ LSB} = FS/4096, \text{ 或} \\ 2.5 \text{ V}/4096 = 0.61 \text{ mV}, \text{ 或} \\ 610 \mu\text{V} (V_{REF} = 2.5 \text{ V})$$

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39所示。

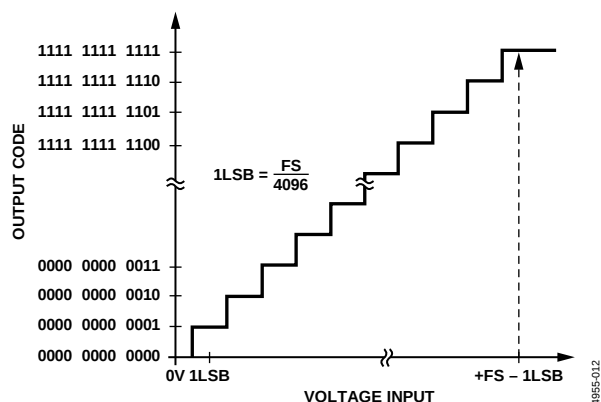


图39. 伪差分或单端模式下ADC传递函数

全差分模式

差分信号的幅值为输入电压引脚 V_{IN+} 和 V_{IN-} 的信号差值(即 $V_{IN+} - V_{IN-}$)。因此, 差分信号的最大幅值为 $-V_{REF}$ 至 $+V_{REF}$ 峰值(即 $2 \times V_{REF}$)。这与共模模式(CM)无关。共模模式是两个输入信号的平均值, 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 $1 \text{ LSB} = 2 V_{REF}/4096$, 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。设计编码转换都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40所示。

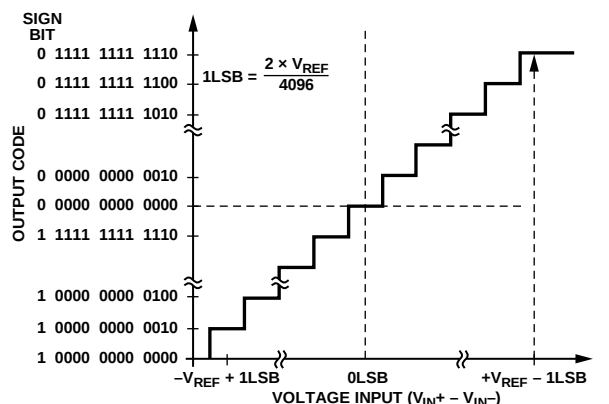


图40. 差分模式下ADC传递函数

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41所示寄存器中的16位至27位，同样地，需要注意的是在全差分模式下，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。

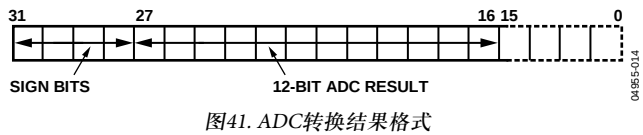


图41. ADC转换结果格式

DACxDAT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640 μ A。使用内部基准电压源时，电流要增加140 μ A。转换过程中，额外电流是0.3 μ A乘以采样频率(单位为kHz)。图33显示了功耗与ADC采样频率的关系。

时序

ADC时序如图42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8个时钟周期，时钟为2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。

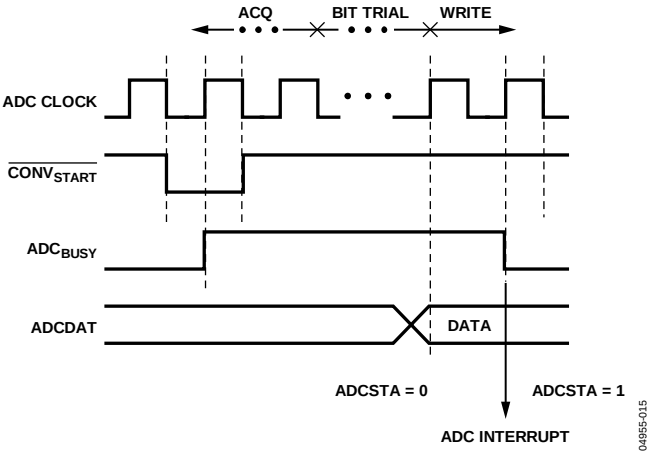


图42. ADC时序

ADuC7019

ADuC7019和ADuC7020相比只是差一个缓冲ADC通道ADC3，另外它只有三个DAC。第四个DAC的输出缓冲在内部连接到ADC3的通道，如图43所示。

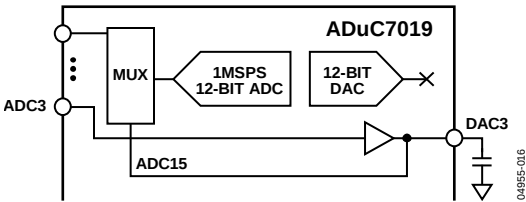


图43. ADC3缓冲输入

注意：DAC3这个输出引脚必须和AGND之间连一个10 nF的电容，并且此通道只能用来测量直流电压。此通道可能需要进行ADC校准。

寄存器接口

这部分我们介绍控制和配置ADC的8个寄存器。

表17. ADCCON寄存器

名称	地址	默认值	访问类型
ADCCON	0xFFFF0500	0x0600	R/W

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、伪差分模式或全差分模式)和转换类型等。该寄存器如表18所示。

表18. ADCCON寄存器位功能描述

位	值	描述
15:13		保留。
12:10		ADC时钟速度。
	000	fADC/1.该分频器用于在外部时钟频率小于41.78 MHz时获得1 MSPS ADC。
	001	fADC/2(默认值)。
	010	fADC/4.
	011	fADC/8.
	100	fADC/16.
	101	fADC/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		保留。
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5 μ s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。
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。
	其它	保留。

表19. ADCCP寄存器

名称	地址	默认值	访问类型
ADCCP	0xFFFF0504	0x00	R/W

ADCCP是ADC正向通道选择寄存器。该寄存器如表20所示。

表20. 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。

表21. ADCCN寄存器

名称	地址	默认值	访问类型
ADCCN	0xFFFF0508	0x01	R/W

ADCCN是一个ADC反向通道选择寄存器。该寄存器如表22所示。

表22. 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	内部基准电压(自诊断特性)。
	其它	保留。

表23. ADCSTA寄存器

名称	地址	默认值	访问类型
ADCSTA	0xFFFF050C	0x00	R

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。如果通过ADCCON寄存器使能，则可以在P0.5引脚输出ADC_{BUSY}的状态(参见“通用输入/输出”部分)。

表24. ADCDAT寄存器

名称	地址	默认值	访问类型
ADCDAT	0xFFFF0510	0x00000000	R

ADCDAT为ADC数据结果寄存器。里面存放12位ADC转换结果数据，如图41所示。

表25. ADCRST寄存器

名称	地址	默认值	访问类型
ADCRST	0xFFFF0514	0x00	R/W

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。

表26. ADCGN寄存器

名称	地址	默认值	访问类型
ADCGN	0xFFFF0530	0x0200	R/W

ADCGN是一个10位增益校准寄存器。

表27. ADCOF寄存器

名称	地址	默认值	访问类型
ADCOF	0xFFFF0534	0x0200	R/W

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。

差分模式

ADuC7019/20/21/22/24/25/26/27/28/29都包含一个基于两个容性DAC的逐次逼近型ADC。图44和图45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44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。

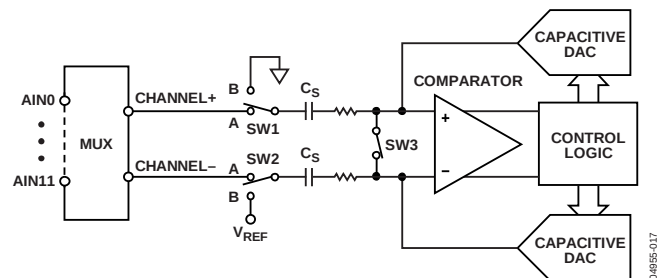


图44. ADC采样阶段

当ADC启动转换，如图45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。注意这里驱动输入电压引脚V_{IN+}和V_{IN-}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。

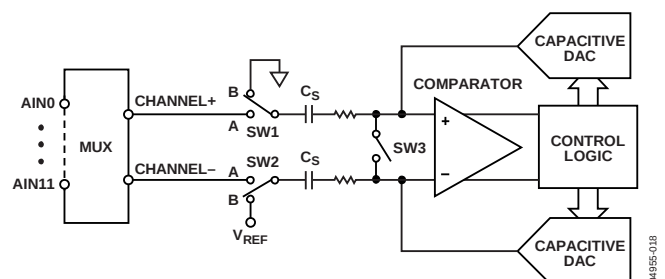


图45. ADC转换阶段

伪差分模式

在伪差分模式中，模拟输入负通道(channel-)连接在ADuC7019/20/21/22/24/25/26/27/28/29的 V_{IN-} 引脚上，SW2开关在A (Channel-)和B (V_{REF})之间进行切换。 V_{IN-} 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} 。

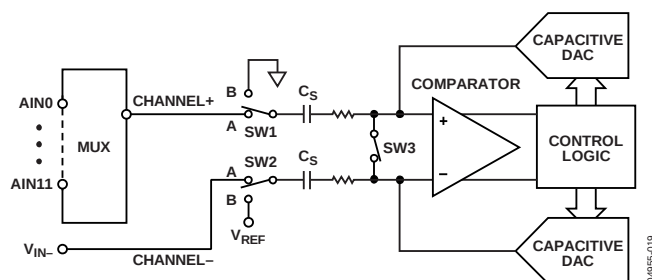


图46. 伪差分模式下ADC

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} 。

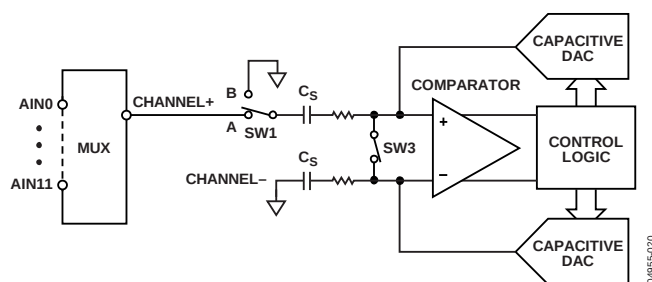


图47. 单端模式下ADC

模拟输入结构

ADC模拟输入结构等效电路如图48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48中，电容C1典型值为4 pF，可基本上被归为引脚寄生电容。电阻是由开关阻抗构成的集总元件。电阻典型值为100 Ω 左右；电容C2为ADC采样电容，典型值为16 pF。

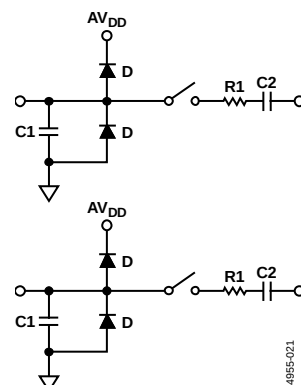


图48. 等效模拟输入电路转换阶段：
开关打开，采样阶段；开关关闭

在交流应用中，建议在相应的模拟输入引脚用一个RC低通滤波器来滤除模拟输入信号的高频成份。在对谐波失真和信噪比要求严格的应用中，模拟输入应采用一个低阻抗源进行驱动。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49和图50为ADC前端的示例。

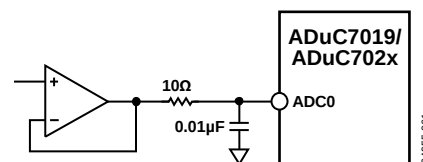


图49. 带缓冲的单端/伪差分输入

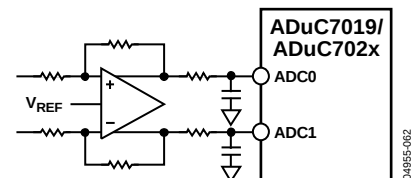


图50. 带缓冲的差分输入

不用放大器来驱动模拟输入端时，应将源阻抗限制在1 k Ω 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28给出了一些计算出的 V_{CM} 最小值和最小值。

表28. 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

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±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±3%。

温度传感器

ADuC7019/20/21/22/24/25/26/27/28/29提供从片内带隙基准电压输出并与绝对温度成比例的电压。该电压可以通过前端ADC多路复用器连接到ADC模拟输入通道，这样就可以很方便的形成一个内部温度传感器通道，用于测量芯片温度，测量精度典型值为±3°C。

以下示例显示了如何使用内部温度传感器：

```
int main(void)
{
float a = 0;
short b;
ADCCON = 0x20;    // power-on the ADC
delay(2000);
```

```
ADCCP = 0x10; // Select Temperature
Sensor as an // input to the ADC

REFCON = 0x01; // connect internal 2.5V
reference // to Vref pin

ADCCON = 0xE4; // continuous conversion
while(1)
{
    while (!ADCSTA){};
    // wait for end of conversion
    b = (ADCDAT >> 16);

    // To calculate temperature in °C, use
the formula:

    a = 0x525 - b;

    // ((Temperature = 0x525 - Sensor
Voltage) / 1.3)

    a /= 1.3;
    b = floor(a);
    printf("Temperature: %d
oC\n",b);
}

return 0;
```

带隙基准电压源

ADuC7019/20/21/22/24/25/26/27/28/29均内置有一个2.5 V的片内带隙基准电压，它可用于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 μF的电容，以确保ADC转换时的稳定性和快速响应。片内基准电压源也可以连接至一个外部引脚(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，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。

表29. REFCON控制寄存器

名称	地址	默认值	访问类型
REFCON	0xFFFF048C	0x00	R/W

带隙基准电压接口包括一个8位REFCON寄存器，如表30所示。

表30. REFCON寄存器位功能描述

位	描述
7:1	保留。
0	内部基准电压输出使能。置1为内部2.5 V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。清0为内部基准电压源断开与V _{REF} 引脚的连接。

非易失性FLASH/EE存储器

ADuC7019/20/21/22/24/25/26/27/28/29片内集成Flash/EE存储器技术，为用户提供可在线重编程的非易失性存储空间。

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ADuC7019/20/21/22/24/25/26/27/28/29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。

每个器件均内置一个64 KB的Flash/EE存储器阵列，其中低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，用于将出厂标定系数下载到各种已校准的外设，如ADC、温度传感器、带隙基准电压源等。用户代码无法访问这2 kB嵌入式固件。

Flash/EE存储器可靠性

器件上的Flash/EE存储器阵列有两个关键的特性：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. 初始页面擦除时序
2. 读/校验时序(单个Flash/EE)
3. 存储器的字节编程时序
4. 二次读/校验时序(耐久性周期)

在进行可靠性验证时，Flash/EE存储器中的三页(顶、中和底)的每半个字(16位宽)可循环存取10000次(从0x0000至0xFFFF)。如表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，如表1所述，Flash/EE存储器循环工作的次数到达指定的耐久性限值，然后确定数据保持特性。这表明每次对Flash/EE存储器进行重新编程时，都保证Flash/EE存储器在完全指定的保持期限内保持数据。此外，注意根据0.6 eV的活化能，保持期限随 T_j 而递减，如图51所示。

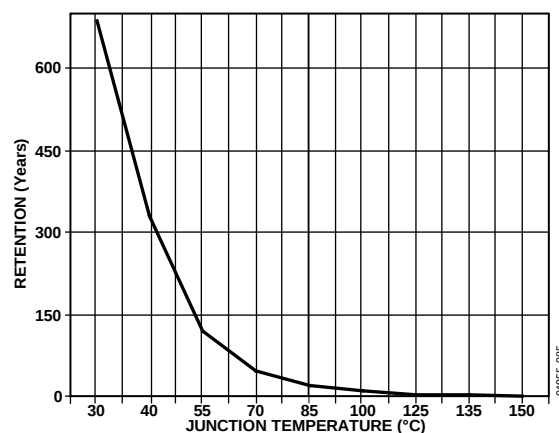


图51. Flash/EE存储器数据保持期限

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)

ADuC7019/20/21/22/24/25/26/27/28/29通过标准的通用异步串口(UART)或I²C端口下载代码。如果BM引脚通过一个1 kΩ的外部下拉电阻下拉到低电平，这样在复位或上电周期后，器件就会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62 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-806](#)说明了通过I²C端口串行下载的协议。

JTAG访问

JTAG协议允许借助片内JTAG接口进行代码下载和调试。

安全性

提供给用户的62 kB Flash/EE存储区域实现了读写保护。

通过配置FEEPRO/FEEHIDE寄存器(参见表42)的第31位，可以保护这62 kB，使其不能通过JTAG编程模式来读取；该寄存器的其他31位可以保护闪存不被写入。每位保护4页，即2 kB。写保护对所有类型的访问都有效。

三种级别保护

- 可通过直接写入FEEHIDE寄存器来设定或取消保护。复位后，保护失效。
- 可通过写入FEEPRO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。FEEPRO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FEEPRO。批量擦除可以将密钥设回0xFFFF，但同时也会擦除所有的用户代码。
- 通过FEEPRO寄存器和一个特定密钥值可以永久保护闪存中的数据：0xDEADDEAD。此时即使再次输入密钥也不允许更改FEEPRO寄存器。

密钥写入时序

1. 写入FEEPRO寄存器的某一位，相应页面会被保护；
2. 将FEEMOD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FEEADR和FEEDAT寄存器；
4. 在FEECON寄存器中运行写密钥命令0x0C，通过监测FEESTA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FEEPRO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)：

```
FEEPRO=0xFFFFFFFF; //Protect pages 4 to 7
FEEMOD=0x48; //Write key enable
FEEADR=0x1234; //16 bit key value
FEEDAT=0x5678; //16 bit key value
FEECON= 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FEEADR = 0xDEAD且FEEDAT = 0xDEAD。

Flash/EE控制接口

串行下载和JTAG编程采用了Flash/EE控制接口，其中包括本节所列的8个寄存器。

表31. FEESTA寄存器

名称	地址	默认值	访问类型
FEESTA	0xFFFFF800	0x20	R

FEESTA是一个只读寄存器，它反映闪存控制接口的状态。如表32所示。

表32. FEESTA寄存器位功能描述

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

表33. FEEMOD寄存器

名称	地址	默认值	访问类型
FEEMOD	0xFFFFF804	0x0000	R/W

FEEMOD用来设置Flash控制接口的工作模式。表34显示FEEMOD寄存器位功能描述。

表34. FEEMOD寄存器位功能描述

位	描述
15:9	保留。
8	保留。该位应该始终清0。
7:5	保留。除了写密钥时，这些位总是清0。参见“密钥写入时序”一节。
4	Flash/EE中断使能。用户置1，使能Flash/EE中断。当处理器执行完一条命令后，将产生中断。用户清0，禁用Flash/EE中断。
3	擦/写命令保护。用户置1，使能擦/写命令；该位清0，禁用擦/写命令。
2:0	保留。这些位总是清0。

表35. FEECON寄存器

名称	地址	默认值	访问类型
FEECON	0xFFFFF808	0x07	R/W

FEECON是一个8位命令寄存器。命令说明如表36所示。

表36. FEECON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	向FEEDAT内载入16位数据。 由FEEADR索引。
0x02 ¹	单次写入	在FEEADR指向的地址内写入FEEDAT。 此操作耗时50 μs。
0x03 ¹	擦/写	擦除由FEEADR索引的存储页，并且把 FEEDAT中的数据写入FEEADR所指的存储 区域。此项操作大约需要24 ms。
0x04 ¹	单次验证	将FEEADR所指地址中的数据与FEEDAT 中的数据进行比较，比较的结果由 FEESTA的第1位显示。
0x05 ¹	单次擦除	擦除由FEEADR索引的页面。
0x06 ¹	批量擦除	擦除62 kB的用户空间。而2 KB的内核 空间是被保护的。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。参见“批量擦除命令执行序列 ”部分。
0x07	保留	保留。
0x08	保留	保留。
0x09	保留	保留。
0x0A	保留	保留。
0x0B	签名	将64 KB Flash/EE的一个签名写入24位 FEESIGN寄存器中。此项操作需要 32,778个时钟周期。
0x0C	保护	该命令只能执行一次。只有批量擦除 命令(0x06)可以将FEEPRO中的值保存 或去除。
0x0D	保留	保留。
0x0E	保留	保留。
0x0F	Ping	无操作；产生中断。

¹在执行完这些命令后，FEECON寄存器总是立即读取0x07。

表37. FEEDAT控制寄存器

名称	地址	默认值	访问类型
FEEDAT	0xFFFFF80C	0xFFFF ¹	R/W

¹X = 0、1、2或3。

FEEDAT是一个16位数据寄存器。

表38. FEEADR寄存器

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

FEEADR是另一个16位地址寄存器。

表39. FEESIGN寄存器

名称	地址	默认值	访问类型
FEESIGN	0xFFFFF818	0xFFFFFFFF	R

FEESIGN为24位编码签名。

表40. FEEPRO寄存器

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

FEEPRO寄存器在寄存器复位后提供保护。它需要一个软件密钥(见表42)。

表41. FEEHIDE寄存器

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

FEEHIDE寄存器提供立即保护。它不需要任何软件密钥。
注意，复位后，FEEHIDE中的保护设置将清0(见表42)。

表42. FEEPRO和FEEHIDE寄存器位功能描述

位	描述
31	读保护位。用户清零，以保护所有代码。 置1，可以读取所有代码。
30:0	123页到120页、119页到116页、0页到3页的写保护。 清0，页写保护。置1，允许进行页写入。

批量擦除命令执行序列

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

SRAM和Flash/EE执行时间

SRAM执行时间

因为一个最小时钟周期为22 ns，而访问SRAM所需的时间为2 ns，所以从SRAM中取指令只需一个时钟周期。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22 ns，所以从Flash/EE执行指令无法在1个周期内完成(当CD位=0时，从SRAM执行指令就能在一个周期内完成)。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，需要两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一个时钟周期。

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43所示。

表43. 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+40 ns。

² N代表使用多重装载/存储指令时，装载或存储数据的个数(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52所示。

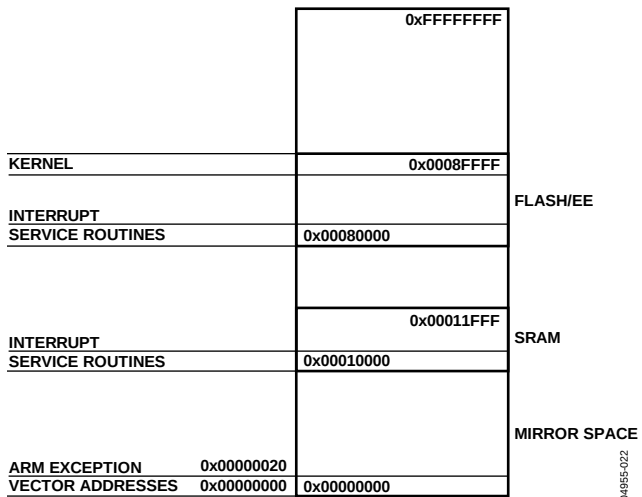


图52. 异常重映射

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。

重映射操作

ADuC7019/20/21/22/24/25/26/27/28/29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，RSTCLR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。

表44. REMAP寄存器

名称	地址	默认值	访问类型
REMAP	0xFFFF0220	0xXX ¹	R/W

¹取决于具体的器件型号。

表45. REMAP寄存器位功能描述

位	名称	描述
4	重映射	只读位。标明Flash/EE存储器可用空间的大小。如果该位置1，表示Flash/EE中只有32 KB空间可用。
3		只读位。标明SRAM存储器可用空间的大小。如果该位置1，表示SRAM中只有4 KB空间可用。
2:1		保留。
0		重映射位。该位置1，映射SRAM到地址0x00000000；.重映射Flash/EE到地址0x00000000后该位自动清0。

表46. RSTSTA寄存器

名称	地址	默认值	访问类型
RSTSTA	0xFFFF0230	0x01	R/W

表47. RSTSTA寄存器位功能描述

位	描述
7:3	保留。
2	软件复位。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。

表48. RSTCLR寄存器

名称	地址	默认值	访问类型
RSTCLR	0xFFFF0234	0x00	W

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

其他模拟外设

DAC

ADuC7019/20/21/22/24/25/26/27/28/29片内集成有2个、3个或4个12位电压输出DAC，具体取决于产品型号。每个DAC都有一个轨到轨电压输出缓冲器，驱动能力为5 k Ω /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50)和DAC0DAT(参见表52)的具体功能。

表49. 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50. DAC0CON寄存器位功能描述

位	名称	值	描述
7:6	DACCLK		保留。
5			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。置1，DAC正常操作使能。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。
		00	掉电模式。DAC输出为三态。
		01	0 V至 DAC_{REF} 范围。
		10	0 V至 V_{REF} （2.5 V）范围。
		11	0 V至 AV_{DD} 范围。

表51. 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52. 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架构由一电阻串DAC和一个输出缓冲放大器构成。功能等效框图如图53所示。

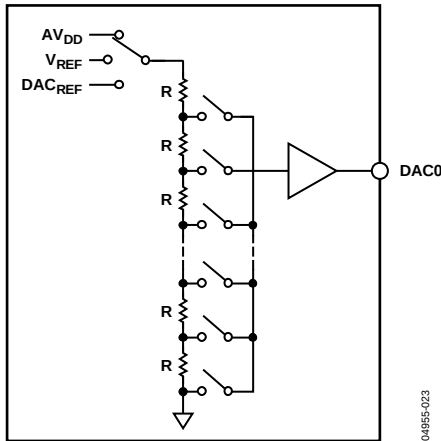


图53. DAC结构

如图53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至 V_{REF} 模式中，DAC输出传递函数范围为0至2.5 V内部基准电压 V_{REF} 。

每个DAC输出缓冲放大器都有一个真轨到轨输出级。也就是说，当输出空载时，DAC输出摆幅能够达到 AV_{DD} 或地电平的5 mV范围以内。此外，当驱动一个5 k Ω 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 AV_{DD} 处的线性降级是由输出放大器的饱和引起的，图54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54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54仅代表输入范围为0至 AV_{DD} 模式时的传递函数。在0至 V_{REF} 或0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。

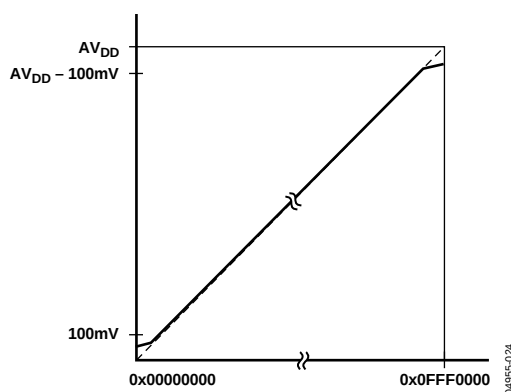


图54. 放大器饱和引起的端点非线性

当有输出负载时，图54中的端点非线性会变得更差。ADuC7019/20/21/22/24/25/26/27/28/29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 k Ω 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54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。

电源监控器

ADuC7019/20/21/22/24/25/26/27/28/29电源监控器调节 IOV_{DD} 电源，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。CMP变为高电平后，该位立即清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。

表53. PSMCON寄存器

名称	地址	默认值	访问类型
PSMCON	0xFFFF0440	0x0008	R/W

表54. 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	Trip point selection bit. 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。CMP变为低电平后，该位会被MicroConverter置1，表明I/O电源电压偏低。PSMI位可用来中断处理器。CMP变为高电平后，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

ADuC7019/20/21/22/24/25/26/27/28/29集成了电压比较器。比较器的正输入端与ADC2引脚复用，而负输入端则有两个：ADC3和DAC0。通过配置，电压比较器的输出可以产生系统中断，直接路由至可编程逻辑阵列，启动ADC转换，或者输出到外部引脚CMP_{OUT}，如图55所示。

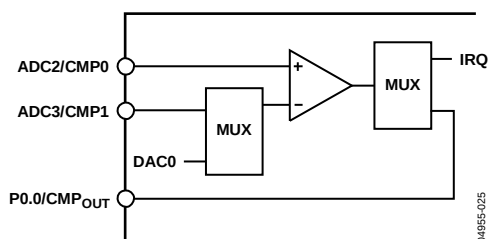


图55. 比较器

注意，因为ADuC7022、ADuC7025和ADu7027不支持DAC0输出，所以把DAC0作为这些器件的比较器输入是不可能的。

迟滞

图56描述了输入失调电压和迟滞的定义方式。

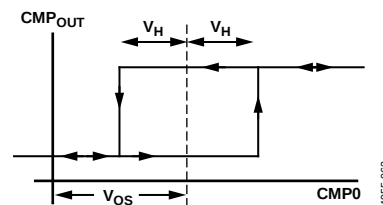


图56. 比较器迟滞传递函数

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)为迟滞范围宽度的 $\frac{1}{2}$ 。

比较器接口

比较器接口由一个16位寄存器CMPCON组成，如表56所示。

表55. CMPCON寄存器

名称	地址	默认值	访问类型
CMPCON	0xFFFF0444	0x0000	R/W

表56. CMPCON寄存器位功能描述

位	名称	值	描述
15:11	CMPEN		保留。
10			比较器使能位。置1，使能比较器；清0，禁用比较器
9:8			
	CMPIN	00	$AV_{DD}/2$ 。
		01	ADC3输入。
		10	DAC0输出。
		11	保留。
7:6	CMPOC		比较器输出配置位。
		00	保留。
		01	保留。
		10	在CMP _{OUT} 上输出。
		11	IRQ。
5	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。
			响应时间。
			对于大信号(2.5 V差分)，响应时间典型值为5 μ s。对于小信号(0.65 mV差分)，响应时间典型值为17 μ s。
4:3	CMPRES	00	典型值为3 μ s。
		11	保留。
		01/10	保留。
2	CMPHYST		比较器迟滞位。将该位置1，有迟滞，大约为7.5 mV；清0，无迟滞。
1	CMPORI		比较器输出上升沿中断。当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

ADuC7019/20/21/22/24/25/26/27/28/29内部集成一个32.768 KHz $\pm$ 3%的振荡器、一个时钟分频器和一个锁相环(PLL)。PLL可以锁住多个(1275)内部振荡器或外部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} 称为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57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。

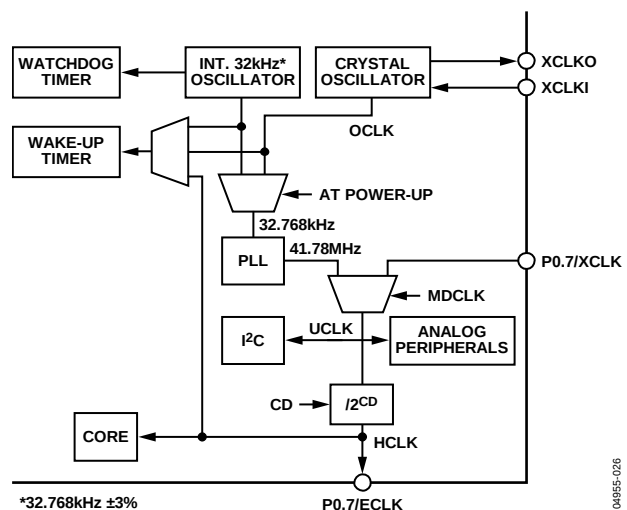


图57. 时钟系统

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POWCON寄存器的正确写序列，强制器件进入浅睡眠(NAP)模式。

当定时器2中断源中断器件，使其离开浅睡眠模式时，时钟源便切换到外部时钟。

源代码示例

```

t2val_old= T2VAL;
T2LD = 5;
TCON = 0x480;

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

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

```

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

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

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44 MHz，容差为1%。

表57. 工作模式¹

模式	内核	外设	PLL	XTAL/T2/T3	IRQ0至IRQ3	启动/上电时间
有效	X	X	X	X	X	130 ms at CD = 0
暂停		X	X	X	X	24 ns (CD = 0); 3 μs (CD = 7)
浅休眠			X	X	X	24 ns (CD = 0); 3 μs (CD = 7)
休眠				X	X	1.58 ms
停止					X	1.7 ms

¹ X表示器件已上电。

表58. 25°C时典型功耗值(单位: mA)

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

源代码示例

```

t2val_old= T2VAL;
T2LD = 5;
TCON = 0x480;

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

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

```

电源控制系统

ADuC7019/20/21/22/24/25/26/27/28/29有多个工作模式可供选择。表57给出了不同模式下各器件的上电情况和上电时间。

表58给出了不同模式下的总功耗(模拟+数字电源电流)在不同时钟分频器位下的一些典型值。ADC关闭。请注意，这些值也包括测量时稳压器和测试板上其他一些器件的功耗。

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61)和POWCON(参见表64)两个MMR控制。PLLCON控制时钟系统的工作模式，而POWCON同时控制内核时钟频率和掉电模式。

为了防止意外编程，写入PLLCON和POWCON寄存器时必须遵循特定的顺序(参见表65)。

表59. PLLKEYx寄存器

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

表60. PLLCON寄存器

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

表61. PLLCON MMR位分配

位	名称	值	描述
7:6 5	OSEL		保留。
			32 kHz PLL输入选择。置1，使用内部32KHz振荡器。默认置1。
4:2 1:0	MDCLK		清0，使用外部32KHz晶振。
			保留。
			时钟模式。
			保留。
		00	保留。
		01	PLL。默认配置。
		10	保留。
		11	P0.7引脚上的外部时钟。

表62. POWKEYx寄存器

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

表63. POWCON寄存器

名称	地址	默认值	访问类型
POWCON	0xFFFF0408	0x0003	R/W

表64. POWCON MMR位分配

位	名称	值	描述
7 6:4	PC		保留。
			工作模式。
		000	活动模式。
		001	暂停模式。
		010	浅休眠。
		011	休眠模式。
		100	IRQ0至IRQ3和定时器2可以唤醒该器件。
		101	停止模式。
		110	IRQ0至IRQ3可以唤醒该器件。
		111	保留。
3 2:0	CD		保留。
			保留。
			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.

表65. PLLCON和POWCON写时序

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

数字外设

三相脉宽调制(PWM)

ADuC7019/20/21/22/24/25/26/27/28/29均内置有一个灵活的、可编程三相脉宽调制(PWM)波形发生器。它可通过编程产生所需的开关信号来驱动三相电压源逆变器用于交流感应马达控制(ACIM)。需要注意的是，只能产生高电平有效的波形。

PWM发生器可以在6个PWM输出引脚(PWM0_H、PWM0_L、PWM1_H、PWM1_L、PWM2_H和PWM2_L)上产生三对PWM信号。6个PWM输出信号包括3个高端驱动信号和3个低端驱动信号。

产生的PWM波形的开关频率和停滞时间可以利用PWM-DAT0和PWMDAT1寄存器编程。此外，三个占空比控制寄存器(PWMCH0、PWMCH1、PWMCH2)直接控制三对PWM信号的占空比。

这六个PWM输出信号中的每一个都由PW MEN寄存器的不同输出使能位控制使能或禁用。此外，PW MEN寄存器有三个控制位允许一个PWM对的两个信号交越。在交越模式下，设定为高侧输出的PWM信号转换为对应的互补低侧输出信号。设定为低端输出的PWM信号转换为对应的高端输出信号。

在许多应用中，需要为开启逆变器功率器件的栅驱动电路提供绝缘隔离。一般情况下，有两种绝缘隔离技术：采用光耦合器的光隔离和采用脉冲变压器的变压器隔离。PWM控制器允许在输出PWM信号时混合一个高频斩波信号，这样更容易与脉冲变压器接口。这种门驱动斩波模式可由PWMCFG寄存器控制。PWMCFG寄存器中有8位直接控制斩波频率。高侧输出和低侧输出也由PWMCFG寄存器的不同使能位分别控制高频斩波率。

PWM可以工作在两种不同的模式下：单次更新模式和二次更新模式。在单次更新模式下，每一个PWM工作周期占空比的值只可以改变一次，因此产生一个关于PWM周期中点对称的PWM波形。在二次更新模式下，可以在一个PWM周期的中点再次更新占空比。

在二次更新模式下，还可以产生一个不对称的PWM波形，这个可以在三相PWM逆变器上能够产生更低次谐波

失真。这种技术允许闭环控制器以更快的速率改变施加于电机绕组上的平均电压。因此，能够产生更快的闭环带宽。PWM模块的工作模式由PWMCON寄存器的一个控制位选择。在单次更新模式下，每个PWM周期都会产生一个PWMSYNC脉冲信号。在二次更新模式下，在每个PWM周期的中点都会产生一个附加的PWMSYNC脉冲。

PWM模块也能在PWM_{SYNC}引脚上提供一个内部同步脉冲，它与PWM开关频率同步。在单次更新模式下，每个PWM周期开始都会产生一个脉冲信号。在二次更新模式下，在每个PWM周期的中点都会产生一个附加的脉冲。脉冲宽度可通过PWMDAT2寄存器编程。PWM模块也能在PWM_{SYNC}引脚上提供一个外部同步脉冲，内外同步的选择在PWMCON寄存器上。SYNC输入时间能够与内部的外设时钟同步，这可通过PWMCON选择。如果该芯片引脚的外部同步脉冲和内部的外设时钟是异步的(典型情况)，则外部的PWMSYNC被认为异步，应同步。从外部脉冲到实际的PWM输出同步逻辑增加延迟和抖动。PWM_{SYNC}引脚产生的脉冲大小必须大于两个内核时钟的周期。

ADuC7019/20/21/22/24/25/26/27/28/29产生的PWM信号可以通过专用的异步PWM关断引脚PWM_{TRIP}停止。当接收到低电平时，PWM_{TRIP}会瞬间使六个PWM输出引脚处于关闭状态(高电平)。由于这个硬件关闭机制是异步的，相关的PWM关闭电路不会通过任何时钟逻辑，这就可以保证即使在内核时钟丢失的情况下PWM也能正确关闭。

用户可通过读PWMSTA寄存器获得PWM系统的状态信息。具体而言，可提供PWM_{TRIP}引脚的状态，同时也有状态位反映PWM是运行在前半周期还是后半周期。

40引脚封装器件

在40引脚封装器件中，PWM的输出不能被直接访问，就如在通用输入/输出部分中所描述的。如下例所示，可通过PLA在GPIO(参见表78)上引出一个通道。

```
PWMCON = 0x1;           // enables PWM o/p
PWMDAT0 = 0x055F;       // PWM switching freq
// Configure Port Pins
GP4CON = 0x300;          // P4.2 as PLA output
GP3CON = 0x1;            // P3.0 configured as
                          // output of PWM0
                          //(internally)

// PWM0 onto P4.2
PLAELM8 = 0x0035;        // P3.0 (PWM output)
                          // input of element 8
PLAELM10 = 0x0059;       // PWM from element 8
```

PWM模块说明

图58显示PWM控制器的功能框图。从引脚PWM0_H到引脚PWM2_L上的六个PWM输出信号由以下四个重要模块控制：

- 三相PWM时钟单元。这是PWM控制器的核心部分，它可以产生三对互补的、死区可调的、基于中心的PWM信号。此单元也能产生内部同步脉冲PWMSYNC，并控制是否使用外部PWM_{SYNC}引脚。
- 输出控制单元。该单元可以调整每一通道的三相时钟单元为高侧输出或为低侧输出。另外，输出控制单元可以单独控制六个PWM输出信号使能或禁用。
- 门驱动单元。该单元可以产生高频斩波以及与PWM信号混合在一起的低频波。
- PWM关闭控制器。该单元可以通过PWM_{TRIP}引脚控制PWM的关闭，并且为时序单元提供准确的复位信号。

PWM控制器由ADuC7019/20/21/22/24/25/26/27/28/29的内核时钟频率驱动，可为ARM核提供两个中断。一个中断在PWM 脉冲出现时产生，另一个在任何一个PWM关闭动作出现时产生。

三相时钟单元

PWM开关频率(PWMDAT0 MMR)

PWM开关频率由PWM周期寄存器PWMDAT0控制。PWM控制器的基本时钟单元为：

t_{CORE} = 1/f_{CORE}

其中，f_{CORE}为微转换器内核频率。

因此，当f_{CORE}为41.78 MHz时，基本时间增量是24 ns。在半个PWM周期以内，写进PWMDAT0寄存器的数据是f_{CORE}时间增量的个数。PWMDAT0的值与要求PWM开关频率(f_{PWM})的函数关系为：

PWMDAT0 = f_{CORE} / (2 × f_{PWM})

因此，PWM开关周期t_s可以写成：

t_s = 2 × PWMDAT0 × t_{CORE}

PWMDAT0 MMR可写入的最大值是0xFFFF = 65535，对应最小的PWM开关频率：

f_{PWM(min)} = 41.78 × 10⁶ / (2 × 65,535) = 318.75 Hz

注意：PWMDAT0的值不能为0和1。

PWM开关死区(PWMDAT1 MMR)

在PWM初始配置时第二个必须设定的重要参数是开关死区，它是在关断一个PWM信号(如0H)与打开其互补信号(0L)之间引入的一个短时延迟。这个短时延迟引入是为了在功率开关关闭(这里为0H)之后到打开互补信号之前保持电路性能。这一时间延迟可以阻止电源转换器直流链路电容引起的潜在破坏性短路。

死区由10位读/写寄存器PWMDAT1控制。只有这一个死区寄存器控制三对PWM输出信号的死区。死区t_D与PWMDAT1寄存器数值之间的关系为：

t_D = PWMDAT1 × 2 × t_{CORE}

因此，如果PWMDAT1的值为0x00A (= 10)，则会在关断一个PWM信号(例如，0H)到打开其互补信号(0L)之间插入一个426 ns的延迟。因此，死区可以2t_{CORE}增量进行编程(或49 ns，在内核时钟41.78 MHz时)。

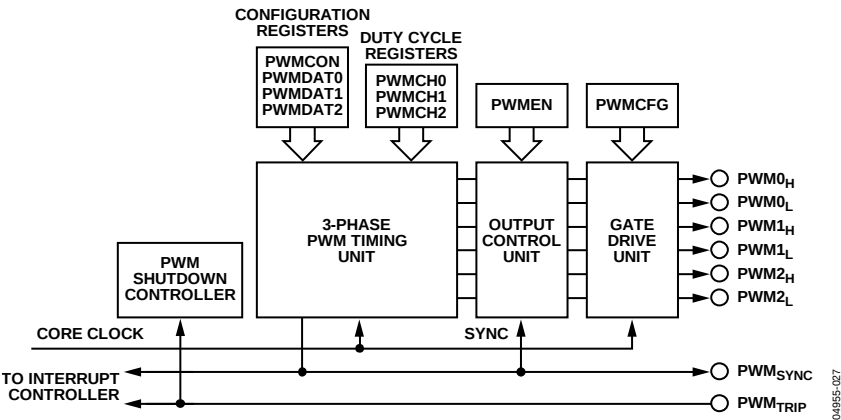


表58. PWM控制器简图

PWMDAT1寄存器是一个10位寄存器，可存储的最大值为0x3FF (= 1023)，对应的最大可编程死区为：

$$t_{D(max)} = 1023 \times 2 \times t_{CORE} = 1023 \times 2 \times 24 \times 10^{-9} = 48.97 \mu s$$

其中，内核时钟为41.78 MHz。

将0写入PWMDAT1寄存器，可以使死区为0。

PWM工作模式(PWMCON和PWMSTA MMR)

如前所述，ADuC7019/20/21/22/24/25/26/27/28/29的PWM控制器可以工作在两种不同的模式下：单次更新模式和二次更新模式。PWM控制器工作在何种模式由PWMCON寄存器的第2位状态决定。如果该位为0，PWM工作在单次更新模式下。如果为1，PWM工作在二次更新模式下。默认为单次更新模式。

在单次更新模式下，每个PWM周期都会产生一个PWMSYNC脉冲信号。该信号上升沿表示一个新的PWM周期开始，并且将PWM配置寄存器(PWMDAT0、PWMDAT1)和PWM占空比寄存器(PWMCH0、PWMCH1和PWMCH2)的新数值锁存到三相时钟单元。此外，在PWMSYNC脉冲的上升沿，PWMCN寄存器也被锁存到输出控制单元，这也就意味着在每一个PWM周期开始时PWM信号的特性和占空比值只可以被改变一次。其结果是左右对称、有关开关周期中点的PWM波形。

在二次更新模式下，在每个PWM周期的中点都会产生一个附加的PWMSYNC脉冲。这个新PWMSYNC脉冲的上升沿被用来锁存PWM配置寄存器、占空比寄存器和PWMCN寄存器新的数值。因此，可以在每一个PWM周期中点时改变特性参数(开关频率、死区)和输出占空比。相应地，还可以产生一个不再关于PWM周期中点对称的PWM开关波形(非对称PWM)。在二次更新模式下，有必要知道当前是运行在PWM的前半周期还是后半周期，此信息由PWMSTA寄存器的第0位反映，它在每个PWM前半周期被清0(初始PWMSYNC脉冲上升沿和二次更新模式产生的新

PWMSYNC脉冲上升沿之间)，在每个PWM后半周期被置1。需要时，这一状态位可用于在PWMSYNC中断服务程序中确定一个特定半周期。

二次更新模式的优点是可产生更低次谐波的电压和更快的控制带宽。然而，在二次更新模式下，对于一个给定的PWM开关频率，PWMSYNC脉冲将以两倍的速度出现。因为在每一个PWMSYNC中断服务程序中都必须重新计算新的占空比值，对于ARM内核来说这是一个比较大的计算量。

PWM占空比(PWMCH0、PWMCH1和PWMCH2 MMR)

引脚PWM0_H至PWM2_L上的六个PWM输出信号的占空比值由三个16位读写占空比寄存器PWMCH0、PWMCH1和PWMCH2控制。占空比寄存器编程为基本时钟单元 t_{CORE} 的整数倍，决定着三相时钟单元产生的期望PWM输出高侧信号超过PWM半个周期的时间。三相时钟单元产生的开关信号也可进行调整，以配合PWMDAT1寄存器中的死区。三相时钟单元根据命令产生动态高电平有效信号来开启相关的功率器件。

图59为在单次更新模式下时钟单元产生的一对典型PWM输出(这里为0H和0L)。图示的所有时间值均为相关寄存器中的整数值，可通过乘以基本时间增量 t_{CORE} 转换为相应的时间。注意在这种模式下开关波形关于每个开关周期的中点对称，这是因为在前半周期和后半周期所用的PWMCH0、PWMDAT0和PWMDAT1值是相同的。

图59图同时也表明了如何调整已编程占空比，以便在一对PWM信号中插入要求的死区。死区的插入是通过将两个PWM信号(0H和0L)开关瞬间从PWMCH0寄存器建立瞬间移除。

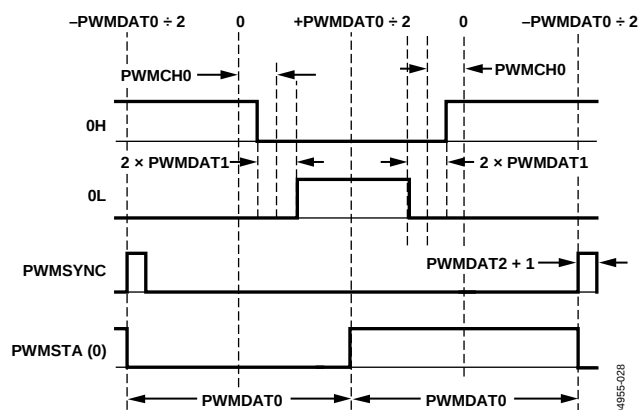


图59. 三相时钟单元的典型PWM输出
(单次更新模式)

为保证输出对称，两个开关沿移动相等的量($PWMDAT1 \times t_{CORE}$)。

图中同时还给出了PWMSYNC脉冲和PWMSTA寄存器的第0位，该位反映当前工作在前半PWM周期还是后半PWM周期。

由时钟单元产生的整个PWM周期(两个半周期)的PWM信号最终开启时间可由下式得到：

在高侧

$$t_{0HH} = PWMDAT0 + 2(PWMCH0 - PWMDAT1) \times t_{CORE}$$

$$t_{0HL} = PWMDAT0 - 2(PWMCH0 - PWMDAT1) \times t_{CORE}$$

对应的占空比(d)为：

$$d_{0H} = t_{0HH}/t_s = \frac{1}{2} + (PWMCH0 - PWMDAT1)/PWMDAT0$$

在低侧

$$t_{0LH} = PWMDAT0 - 2(PWMCH0 + PWMDAT1) \times t_{CORE}$$

$$t_{0LL} = PWMDAT0 + 2(PWMCH0 + PWMDAT1) \times t_{CORE}$$

对应的占空比(d)为：

$$d_{0L} = t_{0LH}/t_s = \frac{1}{2} - (PWMCH0 + PWMDAT1)/PWMDAT0$$

允许的 t_{0H} 和 t_{0L} 最小值为0，此时的占空比为0%。与此相似，最大值为 t_s ，相应的占空比为100%。

图60为二次更新模式下时钟单元的输出信号，显示了一个通用事例，即开关频率、死区和占空比都在PWM后半周期改变。它们其中任何一个或全部三个值既可用于PWM前半周期也可用于PWM后半周期。然而，这并不能保证在二次更新模式下时钟单元产生对称的PWM信号。图60同时显示PWM信号死区的插入与单次更新模式下相同。

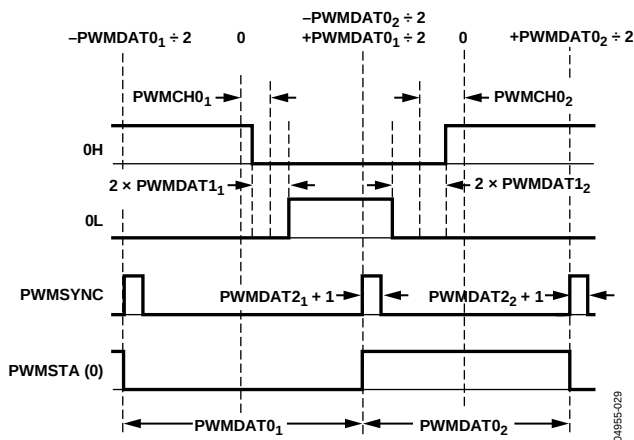


图60. 三相时钟单元的典型PWM输出
(二次更新模式)

总的来说，二次更新模式下PWM信号的开启时间由下式表述：

在高侧

$$t_{0HH} = (PWMDAT0_1/2 + PWMDAT0_2/2 + PWMCH0_1 + PWMCH0_2 - PWMDAT1_1 - PWMDAT1_2) \times t_{CORE}$$

$$t_{0HL} = (PWMDAT0_1/2 + PWMDAT0_2/2 - PWMCH0_1 - PWMCH0_2 + PWMDAT1_1 + PWMDAT1_2) \times t_{CORE}$$

这里下标1表示为前半周期寄存器中的值，下标2表示为后半周期寄存器中的值。

对应的占空比(d)为：

$$d_{0H} = t_{0HH}/t_s = (PWMDAT0_1/2 + PWMDAT0_2/2 + PWMCH0_1 + PWMCH0_2 - PWMDAT1_1 - PWMDAT1_2) / (PWMDAT0_1 + PWMDAT0_2)$$

在低侧

$$t_{0LH} = (PWMDAT0_1/2 + PWMDAT0_2/2 + PWMCH0_1 + PWMCH0_2 + PWMDAT1_1 + PWMDAT1_2) \times t_{CORE}$$

$$t_{0LL} = (PWMDAT0_1/2 + PWMDAT0_2/2 - PWMCH0_1 - PWMCH0_2 - PWMDAT1_1 - PWMDAT1_2) \times t_{CORE}$$

这里下标1表示为前半周期寄存器中的值，下标2表示为后半周期寄存器中的值。

对应的占空比(d)为：

$$d_{0L} = t_{0LH}/t_s = (PWMDAT0_1/2 + PWMDAT0_2/2 + PWMCH0_1 + PWMCH0_2 + PWMDAT1_1 + PWMDAT1_2) / (PWMDAT0_1 + PWMDAT0_2)$$

对于二次更新模式的一个一般实例(见图60)，开关周期表示为：

$$t_s = (PWMDAT0_1 + PWMDAT0_2) \times t_{CORE}$$

同样， t_{0H} 和 t_{0L} 的值被限制在0与 t_s 之间。

通过采用与PWMCH0中介绍的一样的方法对寄存器PWMCH1和PWMCH2进行编程，可以在1H、1L、2H和2L输出上产生与图59和60类似的PWM信号。仅当所有的寄存器PWMDAT0、PWMCH0、PWMCH1和PWMCH2都被至少写入一次，PWM控制器才会产生PWM输出信号。写入这些寄存器时，三相时钟单元的内部计时器即被使能。

写入PWMDAT0寄存器的同时开启PWM主计时器内部计时。如果在初始化时写入PWMCH0、PWMCH1和PWMCH2寄存器之前就先写入PWMDAT0寄存器，在单次更新模式下第一个PWMSYNC脉冲和中断(如果使能)会出现在最初写PWMDAT0寄存器后的 $1.5 \times t_{CORE} \times PWMDAT0$ 秒。在二次更新模式下，第一个PWMSYNC脉冲会在 $PWMDAT0 \times t_{CORE}$ 秒后出现。

输出控制单元

输出控制单元是由一个9位读写寄存器PWMEN控制的，该寄存器控制着输出控制单元的两个主要特性，这些特性可以直接应用于电子计数测量(ECM)控制或二进制/十进制计数测量(BDCM)控制。PWMEN寄存器包括有三个交越位，每位对应一对PWM输出。寄存器第8位置1使能0H/0L这对PWM信号交越，第7位置1使能1H/1L这对PWM信号交越，第6位置1使能2H/2L这对PWM信号交越。对于任何一对PWM信号一旦使能交越模式，时钟单元的高侧PWM信号(例如，0H)就会转换到相应的低侧输出，这样信号最终出现在PWM0_L引脚。当然，对应的时钟单元的低侧输出也会转换到对应的互补高侧输出，这样信号最终会出现在PWM0_H引脚。复位后，三个交越使能位清0，三对PWM信号全部禁用交越模式。PWMEN寄存器有6个控制位(Bit 0至Bit 5)，可用于单独控制六个PWM输出使能或禁用。如果PWMEN寄存器相应的位被置1，那么不管相关的占空比寄存器的值如何，对应的PWM输出就会禁用。PWM输出信号会一直保持关闭状态，直到对应的PWMEN寄存器的使能/禁用位被置1。实施这一输出启用功能后实施交越功能。

复位后，PWMEN寄存器的这六个使能位会被清空，并且所有PWM输出会恢复为默认设置。与占空比寄存器一样，PWMEN在PWMSYNC信号的上升沿被锁定。因此，在单次更新模式下，这个寄存器中值的改变只在每一个PWM周期的开始时有效。在二次更新模式下，PWMEN寄存器的值还可在PWM周期的中点更新。

对于控制一个ECM,在任何时候只有两个转换器引脚可供使用，并且常常是一个引脚连接高侧电平装置的同时另一个引脚连接低侧电平驱动装置。所以，通过对PWM的两个通道设置相同的占空比值(例如PWMCH0 = PWMCH1)以及对PWMEN寄存器的第7位置1来使1H/1L这对PWM信号进行交越，这就有可能在开启A相高侧电平开关的同时开启B相低侧电平开关。在控制ECM时，对于某一些PWM周期，换向器的第三个引脚(这个例子为C相)常常禁用。通过将PWMEN寄存器的第0位和第1位置1以禁用2H和2L PWM输出，可实现该功能。

这种情况如图61所示，在图中可以看到0H和1L信号完全相同，这是因为PWMCH0 = PWMCH1并且B相交越位被置1。

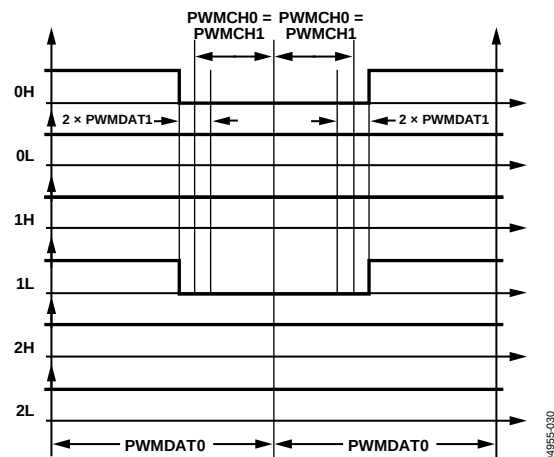


图61. 单次更新模式下用于控制ECM的低电平有效PWM信号(PWMCH0=PWMCH1, 1H和1L对交越, 0L、1H、2H和2L输出禁用)

另外，对PWMEN寄存器的合适的使能/禁用引脚置1可使其它四个信号(0L、1H、2H、和2L)停止工作。在图61中，对PWMEN寄存器合适的值为0x00A7。在ECM正常操作时，需要根据转轴的位置(电机换向)来改变PWMEN寄存器中的值，在这个过程中的一定周期内，换向器的各引脚被禁用。

门驱动单元

PWM控制器的门驱动单元增加的一些特性简化了PWM换向器的相关门驱动电路的设计。如果使用一个变压器耦合，电源装置或一个门驱动放大器，那么有效PWM信号必须在一个高频下截断。16位读写寄存器PWMCFG编程控制这个高频斩波模式。这个斩波有效PWM信号可仅用于高侧驱动，仅用于低侧驱动，或者高侧和低侧开关。所以，在PWMCFG寄存器上有两个不同的控制位单独控制高侧和低侧开关。

图62展示了在高侧和低侧信号使能高频斩波时的典型PWM输出信号。通过将PWMCFG寄存器的第8位置1，可使能高侧PWM输出(0H、1H和2H)斩波。通过将PWMCFG寄存器的第9位置1，可使能低侧PWM输出(0L、1L和2L)斩波。该高斩波频率是通过向PWMCFG寄存器的第0位到第7位写入一个8位字(GDCLK)来控制的。这个高频载波的周期是：

$$t_{CHOP} = (4 \times (GDCLK + 1)) \times t_{CORE}$$

所以这个斩波频率是微处理器内核频率的一个整数分量。

$$f_{CHOP} = f_{CORE} / (4 \times (GDCLK + 1))$$

表70. PWMCFG寄存器

名称	地址	默认值	访问类型
PWMCFG	0xFFFFFC10	0x0000	R/W

PWMCFG is a gate chopping register.

表71. PWMCFG MMR位功能描述

位	名称	描述
15:10		保留。
9	CHOPLO	低侧门斩波使能位。
8	CHOPHI	高侧门斩波使能位。
7:0	GDCLK	PWM门斩波周期(无符号)。

表72. PWMEN寄存器

名称	地址	默认值	访问类型
PWMEN	0xFFFFFC20	0x0000	R/W

PWMEN可使能通道输出和交越。位功能描述见表73。

表73. PWMEN MMR位功能描述

位	名称	描述
8	0H0L_XOVR	通道0输出交越使能位。 置1, 使能通道0输出交越。 清0, 禁用通道0输出交越。
7	1H1L_XOVR	通道1输出交越使能位。 置1, 使能通道1输出交越。 清0, 禁用通道1输出交越。
6	2H2L_XOVR	通道2输出交越使能位。 置1, 使能通道2输出交越。 清0, 禁用通道2输出交越。
5	0L_EN	0L输出使能位。 置1, 禁用PWM的0L输出。 清0, 使能PWM的0L输出。
4	0H_EN	0H输出使能位。 置1, 禁用PWM的0H输出。 清0, 使能PWM的0H输出。
3	1L_EN	1L输出使能位。 置1, 禁用PWM的1L输出。 清0, 使能PWM的1L输出。
2	1H_EN	1H输出使能位。 置1, 禁用PWM的1H输出。 清0, 使能PWM的1H输出。
1	2L_EN	2L输出使能位。 置1, 禁用PWM的2L输出。 清0, 使能PWM的2L输出。
0	2H_EN	2H输出使能位。 置1, 禁用PWM的2H输出。 清0, 使能PWM的2H输出。

表74. PWMDAT0寄存器

名称	地址	默认值	访问类型
PWMDAT0	0xFFFFFC08	0x0000	R/W

PWMDAT0是一个用于开关周期的无符号16位寄存器。

表75. PWMDAT1寄存器

名称	地址	默认值	访问类型
PWMDAT1	0xFFFFFC0C	0x0000	R/W

PWMDAT1是一个用于死区的无符号10位寄存器。

表76. PWMCHx寄存器

名称	地址	默认值	访问类型
PWMCH0	0xFFFFFC14	0x0000	R/W
PWMCH1	0xFFFFFC18	0x0000	R/W
PWMCH2	0xFFFFFC1C	0x0000	R/W

PWMCH0、PWMCH1和PWMCH2是三相的通道占空比寄存器。

表77. PWMDAT2寄存器

名称	地址	默认值	访问类型
PWMDAT2	0xFFFFFC24	0x0000	R/W

PWMDAT2是用于PWM同步脉冲宽度的无符号10位寄存器。

通用输入/输出

ADuC7019/20/21/22/24/25/26/27/28/29有40个通用双向I/O (GPIO)引脚。所有I/O引脚都兼容5V电压, 即GPIO支持5 V输入电压。一般来说, GPIO引脚都有多种功能(表78给出了各引脚功能定义)。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。

当ADuC7019/20/21/22/24/25/26/27/28/29进入省电模式, GPIO引脚会保持它们的状态。

表78. GPIO引脚功能描述

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

¹ 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。

² $\overline{\text{CONV}}_{\text{START}}$ 信号在P2.0的所有模式下均有效。

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

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

GPxPAR是端口0和端口1的参数控制寄存器。注意要先写GPxPAR寄存器，再写GPxDAT寄存器。

表82. GPxPAR寄存器位功能描述

位	描述
31:29	保留。
28	Px.7上拉禁用。
27:25	保留。
24	Px.6上拉禁用。
23:21	保留。
20	Px.5上拉禁用。
19:17	保留。
16	Px.4上拉禁用。
15:13	保留。
12	Px.3上拉禁用。
11:9	保留。
8	Px.2上拉禁用。
7:5	保留。
4	Px.1上拉禁用。
3:1	保留。
0	Px.0上拉禁用。

表83. GPxDAT寄存器

名称	地址	默认值 ¹	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

¹X=0、1、2或3。

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84. GPxDAT寄存器位功能描述

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

表85. GPxSET寄存器

名称	地址	默认值 ¹	访问类型
GP0SET	0xFFFFF424	0x000000XX	W
GP1SET	0xFFFFF434	0x000000XX	W
GP2SET	0xFFFFF444	0x000000XX	W
GP3SET	0xFFFFF454	0x000000XX	W
GP4SET	0xFFFFF464	0x000000XX	W

¹X=0、1、2或3。

GPxSET为端口x的数据设置寄存器。

表86. GPxSET寄存器位功能描述

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

表87. GPxCLR寄存器

名称	地址	默认值 ¹	访问类型
GP0CLR	0xFFFFF428	0x000000XX	W
GP1CLR	0xFFFFF438	0x000000XX	W
GP2CLR	0xFFFFF448	0x000000XX	W
GP3CLR	0xFFFFF458	0x000000XX	W
GP4CLR	0xFFFFF468	0x000000XX	W

¹X=0、1、2或3。

GPxCLR为端口x的数据清除寄存器。

表88. 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。置1，端口x相对应的位被清0，同时GPxDAT寄存器相关的位也被清0。 清0不影响数据输出。
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89描述的具体的I/O功能。

表89. SPM配置

SPMMUX	GPIO (00)	UART (01)	UART/I ² C/SPI (10)	PLA (11)
SPM0	P1.0	SIN	I2C0SCL	PLAI[0]
SPM1	P1.1	SOUT	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	SIN	PLAO[4]
SPM9	P2.0	CONV	SOUT	PLAO[5]

表89详细描述了每一个SPMMUX引脚的模式。这些配置必须通过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。完全兼容16,450串口标准。它不但可以通过串转并的方式从一个外围器件或调制解调器接收数据，而且可以通过并转串的方式从CPU接收数据。它还内置有一个用于发生波特率的小数分频器并且支持网络寻址模式。UART的功能可通过ADuC7019/20/21/22/24/25/26/27/28/29的10个引脚进行设置(如表90所示)。

表90. UART信号描述

引脚	信号	描述
SPM0 (模式1)	SIN	串行接收数据。
SPM1 (模式1)	SOUT	串行发送数据。
SPM2 (模式1)	RTS	发送请求位。
SPM3 (模式1)	CTS	清除发送。
SPM4 (模式1)	RI	响铃指示。
SPM5 (模式1)	DCD	数据载波检测。
SPM6 (模式1)	DSR	数据准备就绪。
SPM7 (模式1)	DTR	数据终端就绪位。
SPM8 (模式2)	SIN	串行接收数据。
SPM9 (模式2)	SOUT	串行发送数据。

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

有两种方式发生UART波特率，普通的450 UART波特率发生器和分数分频器。

普通450 UART波特率生成

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COMDIV0和COMDIV1寄存器中(16位，DL)。

$$\text{波特率} = \frac{41.78 \text{ MHz}}{2^{\text{CD}} - 16 \times 2 \times \text{DL}}$$

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	% Error
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，能够产生范围更宽更精确的波特率。

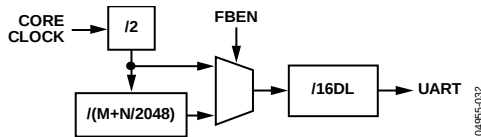


图63. 波特率生成选项

采用小数分频器的波特率计算公式如下：

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

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

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：

$$M + \frac{N}{2048} = \frac{41.78 \text{ MHz}}{19200 \times 2^3 \times 16 \times 8 \times 2}$$

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

其中：

$$M = 1$$

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

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

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19200 bps的波特率时，误差为6.25%。

UART寄存器定义

与UART接口相关的寄存器共有12个：COMTX、COMRX、COMDIV0、COMIEN0、COMDIV1、COMID0、COMCON0、COMCON1、COMSTA0、COMSTA1、COMSCR和COMDIV2。

表92. COMTX寄存器

名称	地址	默认值	访问类型
COMTX	0xFFFF0700	0x00	R/W

COMTX是一个8位发送寄存器。

表93. COMRX寄存器

名称	地址	默认值	访问类型
COMRX	0xFFFF0700	0x00	R

COMRX是一个8位接收寄存器。

表94. COMDIV0寄存器

名称	地址	默认值	访问类型
COMDIV0	0xFFFF0700	0x00	R/W

COMDIV0是一个低字节分频锁存器。COMTX、COMRX和COMDIV0共用相同的地址。当寄存器COMCON0的位7清0时，允许访问寄存器COMTX和COMRX。当COMCON0寄存器的第7位为1时，存取COMDIV0。

表95. COMIEN0寄存器

名称	地址	默认值	访问类型
COMIEN0	0xFFFF0704	0x00	R/W

COMIEN0是一个中断使能寄存器。

表96. COMIEN0 MMR位功能描述

位	名称	描述
7:4	不适用	保留。
3	EDSSI	调制解调器状态中断使能位。置1，在COMSTA1[3:1]中任何一个被置1时产生一个中断；由用户清0。
2	ELSI	Rx状态中断使能位。置0，在COMSTA0[3:1]中任何一个被置1时产生一个中断；由用户清0。
1	ETBEI	使能发送缓冲空中断。置1，在数据发送期间，当发送数据缓冲器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。置1，在数据接收期间，当接收数据缓冲器满时会产生一个中断。由用户清0。

表97. COMDIV1寄存器

名称	地址	默认值	访问类型
COMDIV1	0xFFFF0704	0x00	R/W

COMDIV1是一个高字节分频锁存寄存器。

表98. COMIID0寄存器

名称	地址	默认值	访问类型
COMIID0	0xFFFFF0708	0x01	R

COMIID0为中断识别寄存器。

表99. COMIID0 MMR位功能描述

Bit 2:1 Status Bits	位0 NINT	优先级	定义	清除 操作
00 11	1 0	不适用 1	无中断 接收线路 状态中断	不适用 读取 COMSTA0
10	0	2	接收缓冲 满中断	读取COMRX
01	0	3	发送缓冲 空中断	将数据写入 COMTX或 读取COMIID
00	0	4	调制解调器 状态中断	读取 COMSTA1

表100. COMCON0寄存器

名称	地址	默认值	访问类型
COMCON0	0xFFFFF070C	0x00	R/W

COMCON0是线路控制寄存器。

表101. COMCON0 MMR位功能描述

位	名称	描述
7	DLAB	分频锁存器访问。 用户置1，可以访问寄存器 COMDIV0和COMDIV1。 清0，可以访问COMRX和COMTX寄存器。
6	BRK	设置断开。用户置1，强制SOUT为0。 用户清0，采用正常模式工作。
5	SP	强制奇偶校验。置1，则附加奇偶校验位： 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。用户置1，以发送并检查 奇偶校验位。用户清0，则既不发送也不检查 奇偶校验。
2	STOP	停止位。用户将该位置1，在字长为5位时发送 1.5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。清0，在发送的数据中只 有一个停止位。
1:0	WLS	字长选择： 00 = 5位，01 = 6位， 10 = 7位，11 = 8位。

表102. COMCON1寄存器

名称	地址	默认值	访问类型
COMCON1	0xFFFFF0710	0x00	R/W

COMCON1为调制解调器控制寄存器。

表103. COMCON1 MMR位功能描述

位	名称	描述
7:5 4	LOOPBACK	保留。 回送。用户置1，使能回送模式。 在回送模式下，强制SOUT(见表78)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。将该位清0，以返回正常模式。
3	PEN	奇偶校验使能位。 用户置1，以发送并检查奇偶校验位。 用户清0，则既不发送也不检查奇偶校验。
2	STOP	停止位。用户将该位置1，在字长为5位时 发送1.5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。 清0，在发送的数据中只有一个停止位。
1	RTS	发送请求位。 置1，RTS强制输出0。 清0，RTS强制输出1。
0	DTR	数据终端就绪位。 置1，DTR强制输出0。 清0，DTR强制输出1。

表104. COMSTA0寄存器

名称	地址	默认值	访问类型
COMSTA0	0xFFFFF0714	0x60	R

COMSTA0为线路状态寄存器。

表105. COMSTA0 MMR位功能描述

位	名称	描述
7		保留。
6	TEMT	COMTX和移位寄存器空状态位。 当COMTX和移位寄存器为空时，该位自动置1。 如果COMTX被写入数据，该位自动清0。
5	THRE	COMTX empty.当COMTX为空时，该位自动置1。 如果COMTX被写入数据，该位自动清0。
4	BI	断开错误。 当SIN保持低电平超过最大字长时置1。该位自动清0。
3	FE	帧错误。 当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。
1	OE	溢出错误。如果在读出之前数据被覆盖 该位自动置1。该位自动清0。
0	DR	数据就绪。当COMRX满时，该位自动置1。 读取COMRX后，该位清0。

表106. COMSTA1寄存器

名称	地址	默认值	访问类型
COMSTA1	0xFFFFF0718	0x00	R

COMSTA1为调制解调器状态寄存器。

表107. COMSTA1 MMR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。在上一次读COMSTA1后，如果DCD状态改变了则自动置1。读COMSTA1后自动清0。
2	TERI	下降沿RI。在上一次读COMSTA1后，如果RI由0变为1则置1。读取COMSTA1后，该位自动清0。
1	DDSR	三角波DSR。在上一次读COMSTA1后，如果DSR改变状态则自动置1。读取COMSTA1后，该位自动清0。
0	DCTS	三角波CTS。在上一次读COMSTA1后，如果CTS改变状态则自动置1。读取COMSTA1后，该位自动清0。

表108. COMSCR寄存器

名称	地址	默认值	访问类型
COMSCR	0xFFFF071C	0x00	R/W

COMSCR是一个8位寄存器，用来暂时存储数据。它也可用于网络寻址UART模式。

表109. COMDIV2寄存器

名称	地址	默认值	访问类型
COMDIV2	0xFFFF072C	0x0000	R/W

COMDIV2为16位小数波特率分频寄存器。

表110. COMDIV2 MMR位功能描述

位	名称	描述
15	FBEN	小数波特率发生器使能位。用户置1，使能小数波特率发生器。清0，则使能标准450 UART波特率发生器。
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)。

网络寻址UART模式

这个模式下微转换器可连接到一个有256个节点的串行网络，或者以硬件作为单主，但可以通过软件实现多主连接。COMIEN1寄存器的第7位(ENAM)必须被置位使得UART进入网络寻址模式(参见表112)。注意：在这个模式下不能进行奇偶校验。

网络寻址UART寄存器定义

还有4个寄存器只用于网络寻址UART模式，它们是：COMIEN0、COMIEN1、COMIID1和COMADR。

在网络寻址模式下，COMIEN1寄存器的最低位是网络地址发送控制位。如果该位置1，器件发送地址数据。清0，则发送一般数据。例如：下面就是主机发送从机地址代码的一段程序：

```
COMIEN1 = 0xE7;           //Setting ENAM,
E9BT, E9BR, ETD, NABP

COMTX = 0xA0; // Slave address is 0xA0
while(!(0x020==(COMSTA0 & 0x020))){} //
wait for adr tx to finish.

COMIEN1 = 0xE6;           // Clear NAB bit
to indicate Data is coming

COMTX = 0x55; // Tx data to slave: 0x55
```

表111. COMIEN1寄存器

名称	地址	默认值	访问类型
COMIEN1	0xFFFF0720	0x04	R/W

COMIEN1为8位网络使能寄存器。

表 112. COMIEN1 MMR位功能描述

位	名称	描述
7	ENAM	网络寻址模式使能位。该位置1，进入网络寻址模式。清0，则退出网络寻址模式。
6	E9BT	9位发送使能位。该位置1，使能9位发送，此时ENAM必须被置位。清0，则禁用9位发送。
5	E9BR	9位接收使能位。该位置1，使能9位接收，此时ENAM必须被置位。清0，则禁用9位接收。
4	ENI	网络中断使能位。
3	E9BD	字长。该位置1，数据长度为9位，此时E9BT必须被清0。该位清0，数据长度为8位。
2	ETD	发送引脚驱动使能位。该位置1，在从模式或多主模式下SOUT引脚作为一个输出。清0，SOUT变为三态位。
1	NABP	网络地址位。中断极性位。
0	NAB	地址数据选择位(如果NABP= 1)。该位置1，发送从机的地址。清0，发送数据。

表113. COMIID1寄存器

名称	地址	默认值	访问类型
COMIID1	0xFFFF0724	0x01	R

COMIID1为8位网络中断寄存器。第7位到第4位是保留的(参见表114)。

表114. COMIID1 MMR位功能描述

Bit 3:1 Status Bits	位0 NINT	优先级	定义	清除 操作
000	1		无中断	
110	0	2	匹配网络 地址	读取COMRX
101	0	3	发送地址, 缓冲器空	将数据写入 COMTX或 读取COMIID0
011	0	1	接收线路 状态中断	读取 COMSTA0
010	0	2	接收缓冲 满中断	读取COMRX
001	0	3	发送缓冲 空中断	将数据写入 COMTX或 读取COMIID0
000	0	4	调制解调器 状态中断	读取 COMSTA1

注意，当接收一个网络地址中断时，从机必须保证COMIEN0的第0位被置1，也就是使能接收缓冲器满中断。

表115. COMADR寄存器

名称	地址	默认值	访问类型
COMADR	0xFFFF0728	0xAA	R/W

COMADR是一个8位读/写网络地址寄存器，其中保存的地址由网络寻址UART校验。一旦接收到这个地址，处理器会产生中断并且将COMIID1合适的状态位置1。

串行外设接口

ADuC7019/20/21/22/24/25/26/27/28/29集成了完整的片内硬件串行外设接口(SPI)。SPI是一个工业标准同步串行接口，它允许长为8位数据同步发送和同步接收，也就是说在全双工时比特率最大可达3.48 Mb，如表116所示。SPI接口不是由内核时钟(CD)分频器的控制位操控。在主模式下，POWCON[2:0] = 6或7。

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 (P1.5)、MOSI (P1.6)、SCLK (P1.4)和 $\overline{CS}$ (P1.7)。

在发送侧，SPITX寄存器(和外部的TX移位寄存器)将数据加载到发送引脚(从机模式下为MISO；主机模式下为MOSI)。SPISTA中的发送状态位(Bit 0)指示SPITX寄存器中是否存在有效数据。

类似地，接收数据路径包括SPIRX寄存器(和RX移位寄存器)。SPISTA中的位3指示SPIRX寄存器中是否存在有效数据。如果SPIRX寄存器中的有效数据被覆写，或者RX移位寄存器中的有效数据被丢弃，SPISTA的位5(溢出位)置1。

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)用于同步通过MOSI SCLK周期发送和接收的数据。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：

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

SPI时钟的最大速度取决于时钟分频位并归纳于表116内。

表116. 主模式下SPI速度和时钟分频位对照表

CD位	0	1	2	3	4	5
SPIDIV (16进制)	0x05	0x0B	0x17	0x2F	0x5F	0xBF
SPI速度(MHz)	3.482	1.741	0.870	0.435	0.218	0.109

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。一般在CD = 0时，从机从外部主机接收数据速率可达10.4 Mb。从模式下最大速度的计算公式如下所示：

$$f_{\text{SERIAL CLOCK}} = \frac{f_{\text{HCLK}}}{4}$$

同时主模式和从模式下，数据都在SCL信号的一个沿发送并另一个沿采样。因此，从机时钟的极性和相位必须与主机的配置一致。

片选($\overline{CS}$ 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。

SPI寄存器

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

表117. SPISTA寄存器

名称	地址	默认值	访问类型
SPISTA	0xFFFF0A00	0x00	R

SPISTA是一个8位只读状态寄存器。只有这个寄存器的第1位或第4位产生中断。SPICON寄存器的第6位决定哪一个位产生中断。

表118. SPISTA寄存器位功能描述

位	描述
7:6	保留。
5	SPIRX数据寄存器上溢状态位。 如果SPIRX上溢该位置1。读SPIRX寄存器后清0。
4	SPIRX数据寄存器IRQ。位3或位5置1时自动置1。 读SPIRX寄存器后清0。
3	SPIRX数据寄存器满状态位。 如果SPIRX寄存器中存在一个有效数据则该位自动置1。 读SPIRX寄存器后清0。
2	SPITX数据寄存器下溢状态位。 如果SPITX下溢则该位自动置1。 写入SPITX寄存器后清0。
1	SPITX数据寄存器IRQ。如果该寄存器的第0位被清0或第2位被置1则该位自动置1。向SPITX写入数据或发送完成SPI停止工作时该位清0。
0	SPITX数据寄存器空状态位。 向SPITX寄存器写入数据后置1。 在数据发送过程中这个位始终置1。当SPITX空时清0。

表119. SPIRX寄存器

名称	地址	默认值	访问类型
SPIRX	0xFFFF0A04	0x00	R

SPIRX为8位只读接收寄存器。

表120. SPITX寄存器

名称	地址	默认值	访问类型
SPITX	0xFFFF0A08	0x00	W

SPITX为8位只写发送寄存器。

表121. SPIDIV寄存器

名称	地址	默认值	访问类型
SPIDIV	0xFFFF0A0C	0x1B	R/W

SPIDIV为8位串行时钟分频寄存器。

表122. SPICON寄存器

名称	地址	默认值	访问类型
SPICON	0xFFFF0A10	0x0000	R/W

SPICON为16位控制寄存器。

表123. SPICON寄存器位功能描述

位	描述	功能
15:13	保留	不适用
12	连续传输使能	用户置1可以使能连续发送功能。在主机模式下，数据传输连续进行，直到发送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，直到发送寄存器为空。将该位清0，可禁用连续发送功能。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，那么在一个停转周期后会重新开始发送数据。
11	数据回送使能	将该位置1，以便将MISO连接到MOSI，用于测试软件。将该位清0，以返回正常模式。
10	从机MISO输出使能	将该位置1时，可禁用MISO引脚上的输出驱动器。该位置1后，MISO引脚变为开漏极。将该位清0时，MISO正常工作。
9	片选输出使能	在主模式下置1，禁用片选输出。清0，使能片选输出。 如果也选择了片选输出使能，则SPICON配置为主机之前，应将P1.7配置为 $\overline{CS}$ 。
8	SPIRX上溢覆盖使能	置1，新接收到的串行数据覆盖RX寄存器中的有效数据。 用户清0，则新接收到的串行数据会被丢弃。
7	SPITX下溢模式	置1，发送0；清0，发送前一个数据。
6	传输和中断模式	用户置1，以启动数据传输，同时向SPITX寄存器写入数据。只有当TX为空时产生中断。 清0，读SPIRX寄存器开始发送。只有当RX填满时产生中断。
5	LSB优先传输使能位	用户置1，先发送LSB。用户清0，先发送MSB。
4	保留	
3	串行时钟极性模式位	用户置1，则串行时钟空闲时高电平。用户清0，则串行时钟空闲时低电平。
2	串行时钟相位模式位	用户置1，串行时钟脉冲出现在每一个串行位发送的起始位置。 清0，串行时钟脉冲出现在每一个串行位传输的末尾。
1	主机模式使能位	用户置1，使能主机模式。用户清0，使能从机模式。
0	SPI使能位	用户置1，使能SPI。用户清0，禁用SPI。

I²C兼容接口

ADuC7019/20/21/22/24/25/26/27/28/29支持两个特许的I²C接口。I²C接口既可配置为一个硬件主机，也可配置为一个完整的从机接口。这两个I²C接口完全相同，这里只对I2C0进行详细描述。注意两个主机和一个从机都拥有相互独立的中断(详见“中断系统”部分)。

注意，当被配置为一个I²C主机时，ADuC7019/20/21/22/24/25/26/27/28/29不能产生一个重复起始条件。

有两个GPIO引脚用于数据传输：SDAx和SCLx，它们按照“线与”的方式进行配置，允许在多主机系统下进行仲裁。这两个引脚需要接外部上拉电阻。典型上拉电阻值为10 kΩ。

用户可对I²C总线系统内的I2C总线外设地址进行编程。没有进行传输时，可随时修改这个ID。用户可以对接口进行配置，以使其响应四个从机地址。

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；在起始地址发送期间，主机发送从机的地址和数据发送器的方向。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外设只能配置为主机或从机。同一个I²C通道不能同时支持主机模式和从机模式。

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。

I2C0DIV MMR的波特率的定义如下：

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

其中：

f_{UCLK} = 分频之前的时钟。

DIVH = 时钟的高电平周期。

DIVL = 时钟的低电平周期。

因而，如果希望串行时钟为100 kHz，那么

应该配置 $\text{DIVH} = \text{DIVL} = 0\text{xCF}$

如果希望串行时钟为400 kHz，那么

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

I2CxDIV寄存器对应于DIVH:DIVL。

从机地址

寄存器I2C0ID0、I2C0ID1、I2C0ID2和I2C0ID3包含设备的ID。设备将四个I2C0IDx寄存器中的数据与地址字节做比较。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。

I²C寄存器

I²C外设接口包括本节讨论的18个MMR。

表124. I2CxMSTA寄存器

名称	地址	默认值	访问类型
I2COMSTA	0xFFFF0800	0x00	R/W
I2C1MSTA	0xFFFF0900	0x00	R/W

I2CxMSTA为主机通道状态寄存器。

表125. I2COMSTA MMR位功能描述

位	访问类型	描述
7	R/W	主机发送FIFO清空位。 置1，清空主机发送FIFO。 主机发送FIFO清空后，该位自动清0。 该位也可以清空从机接收FIFO。 主机忙。如果主机忙，该位自动置1。 该位自动清0。
6	R	仲裁失效。在多主机模式下， 如果另一个主机占用总线， 该位置1；如果总线空闲则自动清0。
5	R	不应答NACK。 如果从机没有地址应答，该位自动置1。 读I2COMSTA寄存器后，自动清0。
4	R	主机接收中断请求。 从机接收数据后该位置1。 读I2COMRX寄存器后，自动清0。
3	R	主机发送中断请求。 在一次发送结束时该位置1。 向I2COMTX寄存器写入数据后，自动清0。
2	R	主机发送FIFO下溢。 如果主机发送FIFO下溢，该位自动置1。 向I2COMTX寄存器写入数据后，自动清0。
1	R	主机发送FIFO未溢。 如果从机发送FIFO未溢，该位置1。向I2COSTX寄存器 两次写入数据后，该位自动清0。
0	R	

表126. I2CxSSTA寄存器

名称	地址	默认值	访问类型
I2C0SSTA	0xFFFF0804	0x01	R
I2C1SSTA	0xFFFF0904	0x01	R

I2CxSSTA为从机通道状态寄存器。

表127. I2C0SSTA MMR位功能描述

位	值	描述
31:15		保留。这些位应该写入0。
14		起始解码位。如果器件接收到一个有效的起始条件+地址匹配，则由硬件对该位置1。当产生一个I ² C停止条件或I ² C广播复位时，该位清0。
13		重复起始解码位。如果器件接收到一个有效的重复起始条件+地址匹配，则由硬件对该位置1。当产生一个I ² C停止条件、广播复位或读I2C0SSTA寄存器时该位清0。
12:11		ID解码位。
	00	接收到的地址匹配ID寄存器0。
	01	接收到的地址匹配ID寄存器1。
	10	接收到的地址匹配ID寄存器2。
	11	接收到的地址匹配ID寄存器3。
10		起始和地址匹配中断后停止。在上一个I ² C起始条件+地址匹配后，如果从机接收到一个I ² C停止条件，则由硬件对该位置1。读I2C0SSTA寄存器后该位清0。
9:8		广播ID。
	00	无广播。
	01	广播复位和程序地址。
	10	广播程序地址。
	11	广播匹配可供选择的ID。
7		广播中断。从机接收到任何形式的广播后位置1。对I2CxCFG寄存器的第8位置1后清0。如果是广播复位，所有的寄存器恢复为默认值。如果是硬件广播，接收FIFO会保存广播的第2个字节。除了重编器件地址外与I2C0ALT寄存器相似。更多资料请参阅I ² C总线规范2.1版，2000年1月。
6		从机忙。如果从机忙，该位自动置1。该位自动清0。
5		不应答NACK。主机需要数据却无法获得数据时该位置1。读I2C0SSTA寄存器后该位自动清0。
4		从机接收FIFO溢出。如果从机接收FIFO溢出，该位置1。读I2C0SSTA寄存器后该位自动清0。
3		从机接收中断请求。从机接收数据后该位置1。
2		读I2C0SRX寄存器或清空FIFO后该位自动清0。从机发送中断请求。在一次发送结束时该位置1。向I2C0STX寄存器写入数据后，该位自动清0。
1		从机发送FIFO下溢。如果从机发送FIFO下溢该位置1。向I2C0SSTA寄存器写入数据后该位自动清0。
0		从机发送FIFO未滿。如果从机发送FIFO未滿，该位置1。向I2C0STX寄存器两次写入数据后，该位自动清0。

表128. I2CxSRX寄存器

名称	地址	默认值	访问类型
I2C0SRX	0xFFFFF0808	0x00	R
I2C1SRX	0xFFFFF0908	0x00	R

I2CxSRX为从机通道接收寄存器。

表129. I2CxSTX寄存器

名称	地址	默认值	访问类型
I2C0STX	0xFFFFF080C	0x00	W
I2C1STX	0xFFFFF090C	0x00	W

I2CxSTX为从机通道发送寄存器。

表130. I2CxMRX寄存器

名称	地址	默认值	访问类型
I2C0MRX	0xFFFFF0810	0x00	R
I2C1MRX	0xFFFFF0910	0x00	R

I2CxMRX为主机通道接收寄存器。

表131. I2CxMTX寄存器

名称	地址	默认值	访问类型
I2C0MTX	0xFFFFF0814	0x00	W
I2C1MTX	0xFFFFF0914	0x00	W

I2CxMTX为主机通道发送寄存器。

表132. I2CxCNT寄存器

名称	地址	默认值	访问类型
I2C0CNT	0xFFFFF0818	0x00	R/W
I2C1CNT	0xFFFFF0918	0x00	R/W

I2CxCNT是一个3位主机接收数据计数寄存器。该寄存器中存放的数据为主机将要到从机读取数据的个数，每接收一个字节该寄存器减1。默认时，该寄存器中的值为0，表示接收1字节数据。

Table 133. I2CxADR Registers

名称	地址	默认值	访问类型
I2C0ADR	0xFFFFF081C	0x00	R/W
I2C1ADR	0xFFFFF091C	0x00	R/W

I2CxADR是主机地址字节寄存器。其中存放的数据是主机要通信的器件地址。当主机使能位置位，如果I2CxMTX寄存器中无有效数据，在一个主机传输序列的开始自动发送。

表134. I2CxBYTE寄存器

名称	地址	默认值	访问类型
I2C0BYTE	0xFFFFF0824	0x00	R/W
I2C1BYTE	0xFFFFF0924	0x00	R/W

I2CxBYTE是广播字节寄存器。写入广播字节寄存器的数据不会直接进入发送FIFO。该字节数据一般在地址之前传输序列的开始发送出去。一旦该字节数据被发送并且得到有效应答，I²C总线就会等待另一个数据写入I2CxBYTE或一个地址写入地址寄存器。

表135. I2CxALT寄存器

名称	地址	默认值	访问类型
I2C0ALT	0xFFFFF0828	0x00	R/W
I2C1ALT	0xFFFFF0928	0x00	R/W

I2CxALT是用于从模式的硬件广播ID寄存器。

表136. I2CxCFG寄存器

名称	地址	默认值	访问类型
I2C0CFG	0xFFFFF082C	0x00	R/W
I2C1CFG	0xFFFFF092C	0x00	R/W

I2CxCFG为配置寄存器。

表137. I2C0CFG MMR位功能描述

位	描述
31:5	保留。这些位应该由用户写入0。
14	使能停止中断。置1，在接收到一个停止条件之前、接收到一个有效的起始条件以及地址匹配之后产生中断。清0，在接收到一个停止条件之前不产生中断。
13	保留。
12	保留。
11	时钟延展使能位 (SCL保持低速率)。置1，使能SCL线时钟延展。清0则禁用SCL线时钟延展。
10	保留。
9	从机发送FIFO中断请求使能位。置1，禁用从机发送FIFO中断请求。清0，在读/写位的时钟负脉冲后产生一个中断请求。如果从机发送FIFO空时，用户可以向其中写入数据。在时钟速率为400 kbps并且内核时钟频率为41.78 MHz时，考虑到中断延迟时间，用户有45个时钟周期的时间操作。
8	广播状态位清0。置1，清0广播状态位。在广播状态位清0后由硬件自动清0。
7	主机串行时钟使能位。置1，在主模式使能串行时钟。清0，在主模式禁用串行时钟。
6	回送使能位。置1，内部发送端连接到内部接收端，用于测试用户软件。清0，正常工作。
5	启动延时禁用位。在多主机模式下置1。如果仲裁失效，主机立即尝试再次发送；清0，使能启动延时。在仲裁失效后，主机在尝试再次发送数据前等待。
4	硬件广播使能位。如果该位和第3位置1并且已经接收到一个广播信号(地址为0x00)和1字节数据，器件比较接收寄存器和I2C0ALT中的数据，如果数据匹配，表明器件接收到一个硬件广播。当器件需要紧急呼叫一个主机而又不知道呼叫哪一个时，可使用该功能。这是“对可能相关的主机”的呼叫。ADuC7019/20/21/22/24/25/26/27/28/29寻找这些地址。要求主机注意的器件会将自己的地址嵌入到消息中。所有的主机都会侦听这些消息，然后能够处理该器件要求的主机会与其从机通信并进行相应操作。根据2000年1月的NXP I ² C总线规范，I2C0ALT寄存器的LSB应该始终写入1。
3	广播使能位。通过将该位置1，可以让从机为I ² C广播发送有效应答(ACK)，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，也就是由硬件复位和写从机地址的可编程部分，根据2000年1月的NXP 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。当产生任一广播时，广播中断状态位置1。在复位后用户必须通过设置I ² C接口进行合适的操作。如果接收到的数据字节为0x04，也就是由硬件写子机地址可编程部分，广播中断状态位会建立一个广播，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。
2	保留。
1	主机使能位。该位置1，使能主机I ² C通道。该位清0，禁用主机I ² C通道。
0	从机使能位。该位置1，使能从机I ² C通道，此时会监视从机传输序列中的数据以寻找存放在I2C0ID0、I2C0ID1、I2C0ID2和I2C0ID3中的器件地址。400 kSPS时，内核时钟应以41.78 MHz的频率运行，因为中断延迟一项就可能达到45个时钟周期。I ² C读取位后，用户有0.5个I ² C时钟周期来加载发送FIFO。400 kSPS时，此时间为1.26 μs(中断延迟)。

表138. I2CxDIV寄存器

名称	地址	默认值	访问类型
I2C0DIV	0xFFFF0830	0x1F1F	R/W
I2C1DIV	0xFFFF0930	0x1F1F	R/W

I2CxDIV为时钟分频寄存器。

表139. I2CxIDx寄存器

名称	地址	默认值	访问类型
I2C0ID0	0xFFFF0838	0x00	R/W
I2C0ID1	0xFFFF083C	0x00	R/W
I2C0ID2	0xFFFF0840	0x00	R/W
I2C0ID3	0xFFFF0844	0x00	R/W
I2C1ID0	0xFFFF0938	0x00	R/W
I2C1ID1	0xFFFF093C	0x00	R/W
I2C1ID2	0xFFFF0940	0x00	R/W
I2C1ID3	0xFFFF0944	0x00	R/W

I2CxID0、I2CxID1、I2CxID2和I2CxID3是I2Cx的从机地址设备ID寄存器。

表140. I2CxCCNT寄存器

名称	地址	默认值	访问类型
I2C0CCNT	0xFFFF0848	0x01	R/W
I2C1CCNT	0xFFFF0948	0x01	R/W

I2CxCCNT是8位起始/停止计数寄存器。它可以在起始或停止条件出现时使SDA保持低电平。

表141. I2CxFSTA寄存器

名称	地址	默认值	访问类型
I2C0FSTA	0xFFFF084C	0x0000	R/W
I2C1FSTA	0xFFFF094C	0x0000	R/W

I2CxFSTA为FIFO状态寄存器。

表142. I2C0FSTA MMR位功能描述

位	访问类型	值	描述
15:10	R/W		保留。
9			主机发送FIFO清空位。 置1，清空主机发送FIFO； 当主机发送FIFO已经被清空， 该位自动清0。
8	R/W		该位也可以清空从机接收FIFO。 从机发送FIFO清空位。 置1，清空从机发送FIFO； 从机发送FIFO清空后， 该位自动清0。
7:6		R	主机接收FIFO状态位。 FIFO空。
5:4	R	00	向FIFO写入字节数据。
		01	FIFO中有1字节数据。
		10	FIFO满。
		11	主机发送FIFO状态位。 FIFO空。
3:2	R	00	向FIFO写入字节数据。
		01	FIFO中有1字节数据。
		10	FIFO满。
		11	从机接收FIFO状态位。 FIFO空。
1:0	R	00	向FIFO写入字节数据。
		01	FIFO中有1字节数据。
		10	FIFO满。
		11	从机接收FIFO状态位。 FIFO空。

可编程逻辑阵列(PLA)

每一个ADuC7019/20/21/22/24/25/26/27/28/29都集成有一个完整的可编程逻辑阵列，它由两个相互独立但内部连接的PLA模块组成。每一个模块包括8个PLA单元，所以每种器件共有16个PLA单元。

每个PLA单元都包含有一个双输入的查找表，通过配置可以实现任何基于双输入和一个触发器的逻辑输出功能，如图64所示。

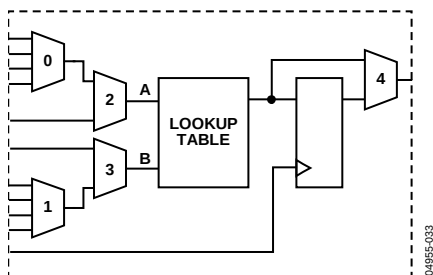


图64. PLA单元

每一个ADuC7019/20/21/22/24/25/26/27/28/29芯片上共有30个GPIO引脚可用于PLA。其中包括16个输入引脚和14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143. 单元输入/输出

PLA模块0			PLA模块1		
单元	输入	输出	单元	输入	输出
0	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

PLA寄存器接口

PLA外设接口包括22个寄存器，以下是对它们的具体描述。

表144. PLAELMx寄存器

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

PLAELMx是单元0到单元15的控制寄存器。通过它们可以配置每一个单元的输入和输出多路复用器、在查找表中选择功能并且选择旁路或使用触发器。参见表145和表150。

表145. 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150)。
8:7		Mux1控制位(见表150)。
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		查找表控制位。
	0000	0。
	0001	或非。
	0010	B与A非。
	0011	A非。
	0100	A与B非。
	0101	B非。
	0110	异或。
	0111	与非。
	1000	与。
	1001	同或。
	1010	B。
	1011	A非或B。
	1100	答：
	1101	A或B非。
	1110	或。
	1111	1。
0		Mux4控制位。
		置1，旁路触发器。
		清0，使用触发器(默认为清0)。

表146. PLACLK寄存器

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

PLACLK是模块0和模块1触发器的时钟选择寄存器。注意当使用GPIO引脚作为PLA模块的时钟输入时，最大频率为44 MHz。

表147. 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	仅外部晶体OCLK (32.768 kHz)。
	101	定时器1溢出。
	其它	保留。
3		保留。
2:0		模块0时钟源选择。
	000	P0.5引脚连接的GPIO时钟。
	001	P0.0引脚连接的GPIO时钟。
	010	P0.7引脚连接的GPIO时钟。
	011	HCLK.
	100	仅外部晶体OCLK (32.768 kHz)。
	101	定时器1溢出。
	其它	保留。

表148. PLAIRQ寄存器

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

PLAIRQ可以使能IRQ0和/或IRQ1和选择IRQ中断源。

表149. 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150. 反馈配置

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

表151. PLAADC寄存器

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

PLAADC是配置PLA作为ADC起始转换信号来源的寄存器。

表152. 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。

表153. PLADIN寄存器

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

PLADIN是一个PLA的数据输入寄存器。

表154. 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。

表155. PLADOUT寄存器

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

PLADOUT是一个PLA的数据输出寄存器。这个寄存器是始终更新的。

表156. 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。

表157. PLALCK寄存器

名称	地址	默认值	访问类型
PLALCK	0xFFFF0B54	0x00	W

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

ADuC7019/20/21/22/24/25/26/27/28/29共有23个中断源，它们都由中断控制器控制。大多数中断都是由片内外围设备产生，例如ADC和UART。四个额外的中断源由外部中断请求引脚IRQ0、IRQ1、IRQ2和IRQ3产生。ARM7TDMI CPU内核只能识别以下两类中断：正常中断请求(IRQ)和快速中断请求(FIQ)。所有中断都可以被单独屏蔽。

通过9个与中断有关的寄存器来管理中断系统的控制和配置，4个用于控制IRQ，4个用于控制FIQ，还有一个MMR用于选择编程中断源。每一个IRQ和FIQ寄存器中的控制位(除了位23)都代表相同的中断源，如表158所示。

表158. IRQ/FIQ寄存器位功能描述

位	描述
0	所有中断的逻辑或(限FIQ)
1	SWI
2	定时器0
3	定时器1
4	唤醒定时器(定时器2)
5	看门狗定时器(定时器3)
6	闪存控制
7	ADC通道
8	PLL锁定
9	I2C0从机
10	I2C0主机
11	I2C1主机
12	SPI从机
13	SPI主机
14	UART
15	外部IRQ0
16	比较器
17	PSM
18	外部IRQ1
19	PLA IRQ0
20	PLA IRQ1
21	外部IRQ2
22	外部IRQ3
23	PWM触发(仅IRQ)/PWM同步(仅FIQ)

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服务。

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

表159. IRQSTA寄存器

名称	地址	默认值	访问类型
IRQSTA	0xFFFF0000	0x00000000	R

IRQSTA(只读寄存器)提供当前使能的IRQ源状态。当置1时，这个源应该向ARM7TDMI内核产生一个有效IRQ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。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。

表160. IRQSIG寄存器

名称	地址	默认值	访问类型
IRQSIG	0xFFFF0004	0x00XXX000 ¹	R

¹ X表示未定义值。

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。

表161. IRQEN寄存器

名称	地址	默认值	访问类型
IRQEN	0xFFFF0008	0x00000000	R/W

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

注意：如果要想取消一个已经使能的中断源，用户必须通过对IRQCLR中相应的位置1来实现。将这个中断源在IRQEN中相应的位清0不会禁用这个中断。

表162. IRQCLR寄存器

名称	地址	默认值	访问类型
IRQCLR	0xFFFF000C	0x00000000	W

IRQCLR(只写寄存器)用于清除IRQEN寄存器的相应位，以屏蔽中断源。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。

FIQ

快速中断请求(FIQ)是进入处理器FIQ模式的一个异常信号。提供此信号的目的是以低延迟处理数据传输或通信通道任务。FIQ接口与IRQ接口相同，但它会提供二级中断(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。

表163. FIQSTA寄存器

名称	地址	默认值	访问类型
FIQSTA	0xFFFFF0100	0x00000000	R

表164. FIQSIG寄存器

名称	地址	默认值	访问类型
FIQSIG	0xFFFFF0104	0x00XXX000 ¹	R

¹X表示未定义值。

表165. FIQEN寄存器

名称	地址	默认值	访问类型
FIQEN	0xFFFFF0108	0x00000000	R/W

表166. FIQCLR寄存器

名称	地址	默认值	访问类型
FIQCLR	0xFFFFF010C	0x00000000	W

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IRQEN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会使IRQEN中的同一位被清0。同样,IRQEN中的某一位被置1会使FIQEN中同一位被清0。一个中断源可以被IRQEN屏蔽和FIQEN屏蔽同时禁用。

注意若要取消一个已经使能的FIQ中断源，用户必须将FIQCLR寄存器中相应的位置1。将一个中断在FIQEN寄存器中相应的位清0不会禁用这个中断。

可编程中断

因为可编程中断是无法屏蔽的，所以它们由另外一个寄存器SWICFG控制，通过这个寄存器可以同时写入IRQSTA和IRQSIG寄存器，和/或FIQSTA和FIQSIG寄存器。32位SWICFG寄存器专门用于软件中断(见表168)。这个存储器映射寄存器允许对可编程中断源进行控制。

表167. SWICFG寄存器

名称	地址	默认值	访问类型
SWICFG	0xFFFFF0010	0x00000000	W

表168. 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都要至少等于中断延迟时间，这样才能保证被中断控制器和用户在IRQSTA/FIQSTA寄存器中检测到。

定时器

ADuC7019/20/21/22/24/25/26/27/28/29有四个通用定时器/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0或满量程，然后再以该寄存器中的值为起始值，重新开始计数。

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

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

如果将定时器设置为递增计数，那么

$$Interval = \frac{(Fs - TxLD) \times Prescaler}{SourceClock}$$

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。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可能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

若要使用“时:分:秒:百分数”格式，应选择32,768 kHz时钟和预分频值256。百位数字段不代表毫秒，而是一秒的1/128 (256/32,768)。代表时、分、秒的位在寄存器中不是连续的。使用TxCON[5:4]设置的“时:分:秒:百位数”格式时，这一配置适用于TxLD和TxVAL。详情参见表169。

表169. 时:分:秒:百分数格式

位	值	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

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，工作时递减计数(如图65所示)。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65所示。

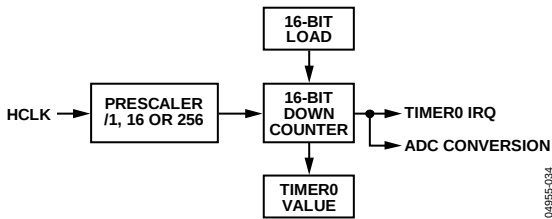


表65. 定时器0框图

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

表170. TOLD寄存器

名称	地址	默认值	访问类型
TOLD	0xFFFF0300	0x0000	R/W

TOLD是一个16位装载寄存器。

表171. T0VAL寄存器

名称	地址	默认值	访问类型
T0VAL	0xFFFF0304	0xFFFF	R

T0VAL是一个16位只读寄存器，用于表示计数器的当前状态。

表172. T0CON寄存器

名称	地址	默认值	访问类型
T0CON	0xFFFF0308	0x0000	R/W

T0CON是一个配置寄存器，如表173所示。

表173. T0CON寄存器位功能描述

位	值	描述
15:8		保留。
7		定时器0使能位。用户置1，使能定时器0。清0，禁用定时器0(默认为0)。
6		定时器0模式。用户置1，选择周期模式。清0，选择自由模式。默认模式。
5:4		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。

表174. T0CLRI寄存器

名称	地址	默认值	访问类型
T0CLRI	0xFFFF030C	0xFF	W

T0CLRI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。它的时钟源可以是32 KHz的外部晶振、内核时钟频率或者是一个外部GPIO(P1.0或P0.6)。时钟输入的最高频率为44 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。这一特性可用来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，如框图66所示。

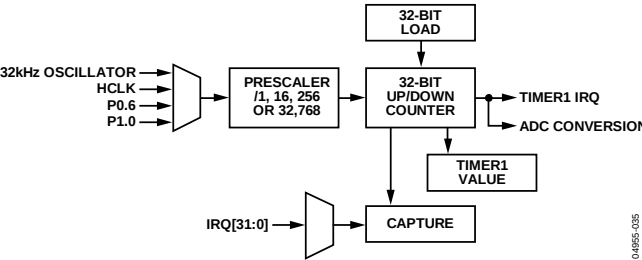


图66. 定时器1框图

定时器1接口有5个寄存器：T1LD、T1VAL、T1CON、T1CLRI和T1CAP。

表175. T1LD寄存器

名称	地址	默认值	访问类型
T1LD	0xFFFF0320	0x00000000	R/W

T1LD是一个32位装载寄存器。

表176. T1VAL寄存器

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

T1VAL是一个32位只读寄存器，用于表示计数器的当前状态。

表177. T1CON寄存器

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

T1CON是一个配置寄存器，如表178所示。

表178. 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，可禁用对事件的实时捕捉。
16:12		事件选择范围0至31。 这些事件如表158所示， 其中所有事件的序号对于定时器1都 相应减少两位。例如，表158中的事件2 对于定时器1来说就是事件0。
11:9		时钟选择。
	000	内核时钟(HCLK)。
	001	外部32.768 kHz晶振。
	010	P1.0上升沿触发。
	011	P0.6上升沿触发。
8		递增计数。 用户置1，定时器1递增计数。
		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。
		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。
		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。

表179. T1CLRI寄存器

名称	地址	默认值	访问类型
T1CLRI	0xFFFF032C	0xFF	W

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。

表180. T1CAP寄存器

名称	地址	默认值	访问类型
T1CAP	0xFFFF0330	0x00000000	R/W

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，工作时可递增计数或递减计数。它的时钟源可以是32 KHz的外部晶振、内核时钟频率或内部32 KHz的振荡器。定时器2的时钟是其所选时钟源的一个分频，分频方式共有1、1/16、1/256、1/32768四种。当内核时钟被禁用时，定时器2仍会继续运行。

计数器可以是标准的32位数模式或下面的格式：时：分：秒：百分之一秒。

定时器2可用于启动ADC转换，如框图67所示。

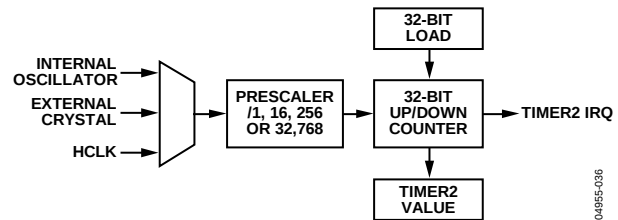


图67. 定时器2框图

定时器2接口有4个寄存器：T2LD、T2VAL、T2CON和T2CLRI。

表181. T2LD寄存器

名称	地址	默认值	访问类型
T2LD	0xFFFF0340	0x00000000	R/W

T2LD是一个32位装载寄存器。

表182. T2VAL寄存器

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

T2VAL是一个32位只读寄存器，用于表示计数器的当前状态。

表183. T2CON寄存器

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

T2CON是一个配置MMR，如表184所示。

表184. T2CON寄存器位功能描述

位	值	描述
31:11		保留。
10:9		时钟源。
	00	外部晶振。
	01	外部晶振。
	10	内部振荡器。
	11	内核时钟(41 MHz/2 ^{CD})。
8		递增计数。
		用户置1, 定时器2递增计数。
		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。
		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。
		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(默认)。
	0100	时钟源/16。
	1000	时钟源/256
		(预计用于数据格式2和3)。
	1111	时钟源/32,768。

表185. T2CLRI寄存器

名称	地址	默认值	访问类型
T2CLRI	0xFFFF034C	0xFF	W

T2CLRI是一个8位寄存器。向其中写入任意值就可以清除定时器2中断。

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(32 KHz)，其时钟分频系数共有1、16、256三种(如图68所示)。

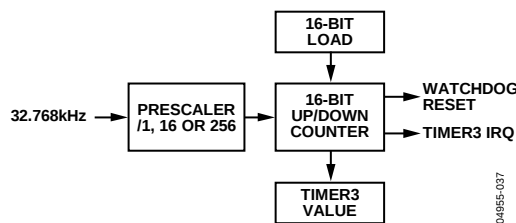


图68. 定时器3框图

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以T3LD寄存器中的数为起始值开始递减计数，一直到0为止，T3LD作为超时定时器。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32 KHz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T3CLRI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T3CLRI。

表186. T3LD寄存器

名称	地址	默认值	访问类型
T3LD	0xFFFF0360	0x0000	R/W

T3LD是一个16位装载寄存器。

表187. T3VAL寄存器

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

T3VAL是一个16位只读寄存器，它记录的是计数器当前的状态。

表188. T3CON寄存器

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

T3CON是配置MMR，如表189所示。

表189. T3CON MMR位功能描述

位	值	描述
15:9		保留。
8		递增计数。用户置1，可将定时器3的计数方式设定为递增计数。清0，定时器3递减计数(默认)。
7		定时器3使能位。用户置1，可启用定时器3。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。清0，选择自由模式。默认模式。
5		看门狗模式使能位。置1，使能看门狗模式。清0，禁用看门狗模式(默认)。
4		安全清除位。置1，使能安全清除。该位清0，禁用安全清除(默认)。
3:2	00	预分频选择位。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。置1，当看门狗定时器递减到0时产生一个IRQ而不是复位。用户清0，禁用IRQ选项。
0		保留。

表190. T3CLRI寄存器

名称	地址	默认值	访问类型
T3CLRI	0xFFFFF036C	0x00	W

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69所示。

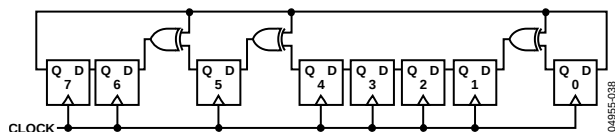


图69. 8位LFSR

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的值不能被访问，且必须在软件中产生和跟踪。

下面是一个示例序列：

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

ADuC7026和ADuC7027是这一系列芯片中唯一拥有外部存储器接口的两个型号。外部存储器接口需要大量的引脚。所以这种接口只能存在于引脚数多的封装形式的芯片上。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191所示。

表191. 外部存储器接口引脚

引脚	功能
AD[16:1]	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
BHE, BLE	字节写入功能

共有4个可用的外部存储区域，如表192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70示例)。这4个区域可单独配置。

表192. 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。

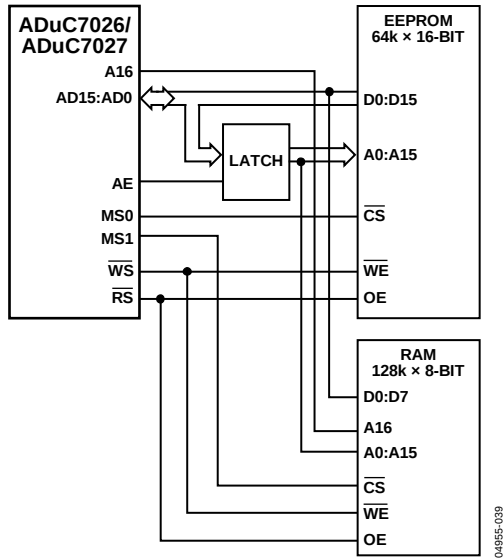


图70. 外部EEPROM/RAM接口

表193. XMCFG寄存器

名称	地址	默认值	访问类型
XMCFG	0xFFFFF000	0x00	R/W

将XMCFG寄存器置1可以使能访问外部存储器。必须在将任何端口引脚配置为外部存储器访问引脚之前对这个寄存器置1。端口引脚也必须通过GPxCON存储器映射寄存器单独使能。

表194. 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195. XMxCON寄存器位功能描述

位	描述
1	选择数据总线宽度。 置1，选择16位数据总线。清0，选择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196. 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197. 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位存储器共享同一个存储区域时。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。
9	清0，在读和写时使能1个周期的地址保持时间。
8	读数据时额外总线转换时间。 置1，禁用额外总线转换时间。
7:4	清0，在读选通(RS)之前和之后使能1个额外时钟。
3:0	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。
	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)。
	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71、图72、图73和图74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。

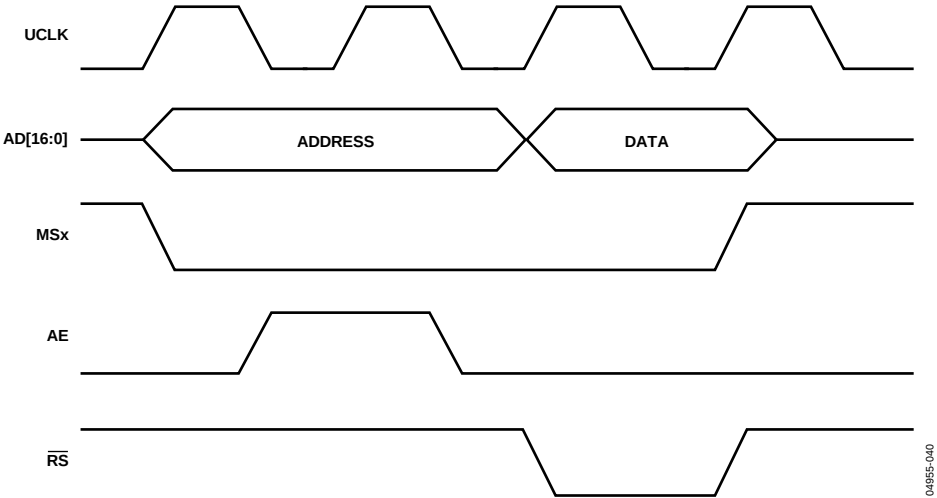


图71 外部存储器读周期

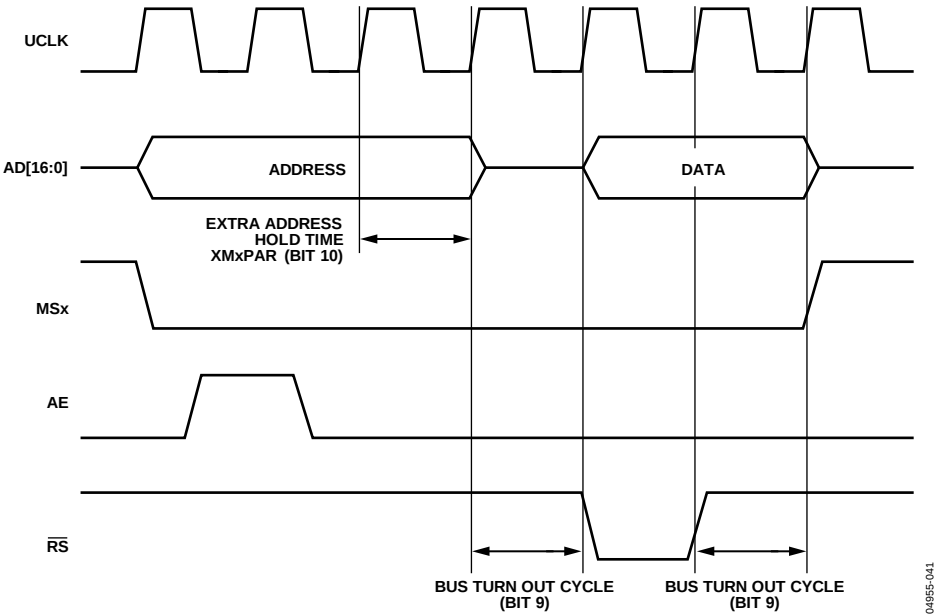


图72. 带有地址保持和总线转换周期的外部存储器读周期

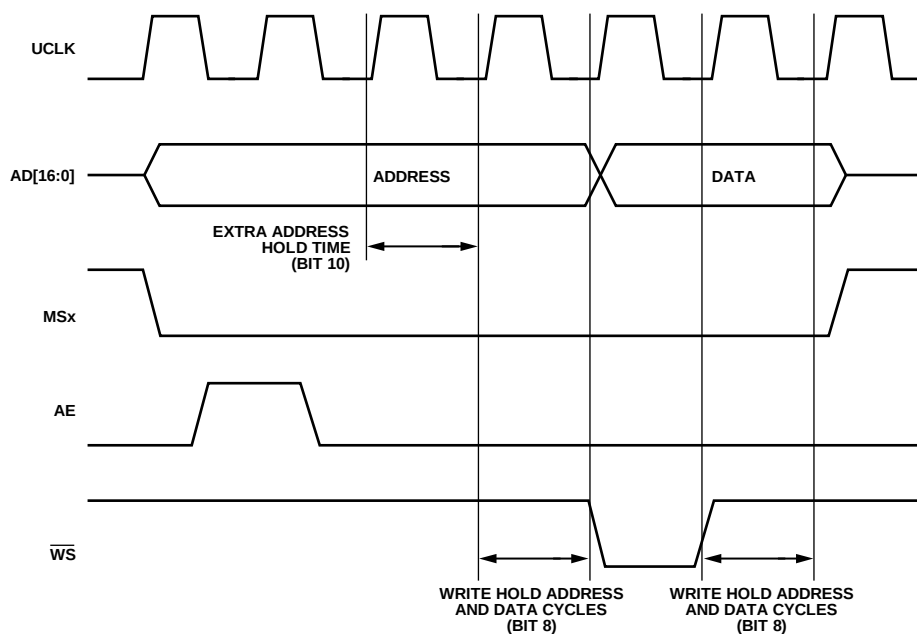


图73. 带有地址和写保持周期的外部存储器写周期

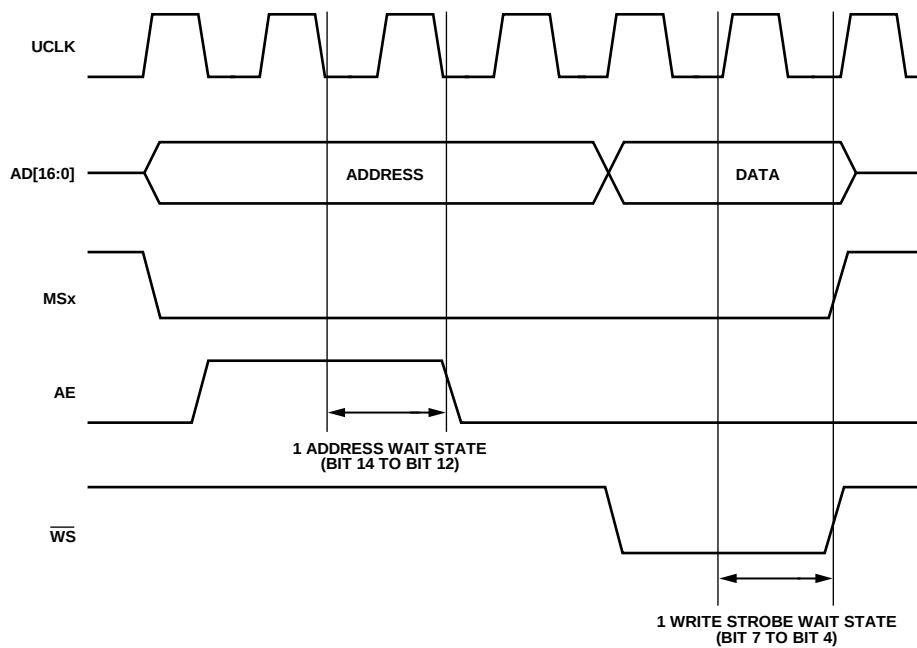


图74. 带有等待状态的外部存储器写周期

硬件设计考虑

电源

ADuC7019/20/21/22/24/25/26/27/28/29工作电压范围为2.7 V至3.6 V。分离的模拟和数字电源引脚(分别为 AV_{DD} 和 IOV_{DD})使得 AV_{DD} 不受 IOV_{DD} 上数字信号干扰的影响。在这种模式下,器件可以在分离电源下工作;也就是说,各个电源的电压可以是不同的。例如,系统的 IOV_{DD} 工作电压为3.3 V,而 AV_{DD} 电压为3 V;反之亦然。图75是一个典型的分离电源配置。

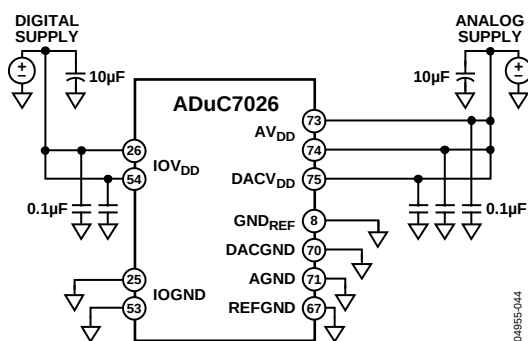


图75. 外部双电源连接

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76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。

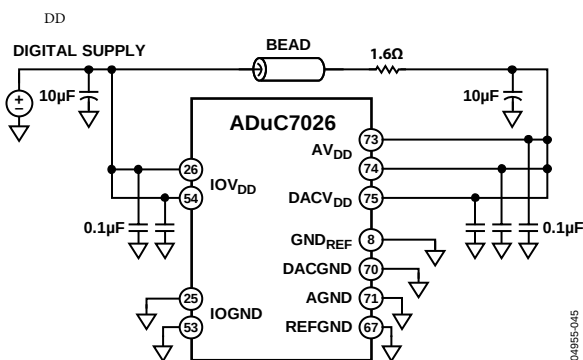


图76. 外部单电源连接

注意:在图75和图76中,在 IOV_{DD} 处有一个大容量值(10 μF)的储能电容,而在 AV_{DD} 处则单独有一个10 μF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 μF)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。

最后,应注意在任何时候ADuC7019/20/21/22/24/25/26/27/28/29的模拟地和数字地引脚必须参考同一个系统地参考点。

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77所推荐的滤波电路。

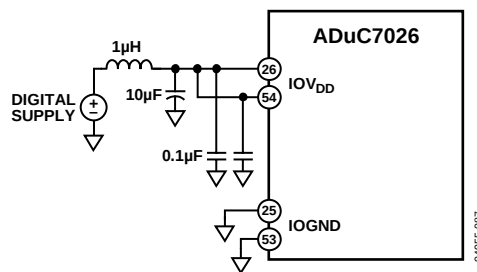


图77. 推荐的 IOV_{DD} 电源滤波电路

线性稳压器

各ADuC7019/20/21/22/24/25/26/27/28/29都需要一个3.3 V单电源,但是内核逻辑需要一个2.6 V的电源。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 μF的补偿电容(应尽量靠近这些引脚)作为电荷槽,如图78所示。

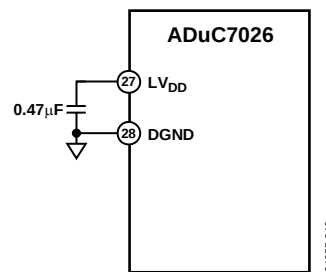


图78. 稳压器连接

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。

接地和电路板布局建议

对于所有的高分辨率数据转换器，为了从ADC和DAC获得最优的性能应特别注意基于ADuC7019/20/21/22/24/25/26/27/28/29的接地和PCB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79a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。在这些例子中，ADuC7019/20/21/22/24/25/26/27/28/29的所有AGND和IOGND引脚都连接到模拟地平面，如图79b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)。

ADuC7019/20/21/22/24/25/26/27/28/29可以放置在数字和模拟部分之间，如图79c所示。

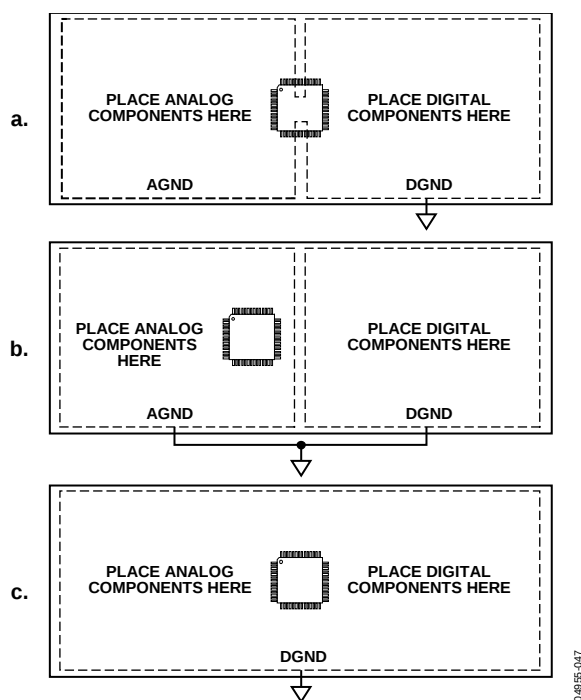


图79. 系统地框图

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79b所示)，因为这会导致IOV_{DD}回路电流强行通过AGND。如果

电路板的左半平面放置了一个带有噪声的数字芯片(如图79c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ADuC7019/20/21/22/24/25/26/27/28/29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5ns)时，应该在每一条相关的线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

ADuC7019/20/21/22/24/25/26/27/28/29的时钟源可以由内部锁相环或者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80所示。这个晶体使得锁相环可以正确锁相进而产生41.78 MHz频率的时钟信号。如果不使用外部晶体，内部振荡器会产生一个41.78 MHz $\pm$ 3%的典型频率。

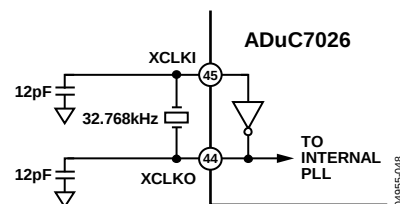


图80. 外部并行谐振晶体连接

如果使用一个外部时钟源输入代替锁相环(如图81所示)，PLLCON寄存器的位1和位0都需要修改，外部时钟从P0.7和XCLK引脚输入。

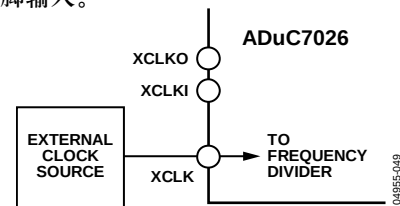


图81. 连接一个外部时钟源

当使用外部时钟源时，ADuC7019/20/21/22/24/25/26/27/28/29的额定时钟频率范围为50 kHz到44 MHz $\pm$ 1%，这可以确保模拟外设和Flash/EE正常工作。

Rev. E | Page 90 of 96

开发工具

基于PC工具

ADuC7019/20/21/22/24/25/26/27/28/29系列有四类开发系统可用：

- ADuC7026 QuickStart Plus是为那些想拥有一个全面硬件开发环境的新用户准备的。因为ADuC7026包括的功能是ADuC7019/20/21/22/24/25/26/27/28/29系列中最全的，它适合于那些希望在这一系列器件中的任何一个上进行开发的用户。所有的器件都是代码兼容的。
- ADuC7020、ADuC7024和ADuC7026 QuickStart系统适合那些已经拥有一个仿真器的用户。

这些系统由以下基于PC(兼容Windows®)的硬件和软件开发工具组成。

硬件

- ADuC7019/20/21/22/24/25/26/27/28/29评估板
- 串行端口编程电缆
- RDI兼容JTAG仿真器(仅包括在ADuC7026 QuickStart Plus中)

软件

- 集成开发环境，包括汇编器、编译器和基于JTAG的非介入式调试器
- 串行下载软件
- 示例代码

其它

CD-ROM文档

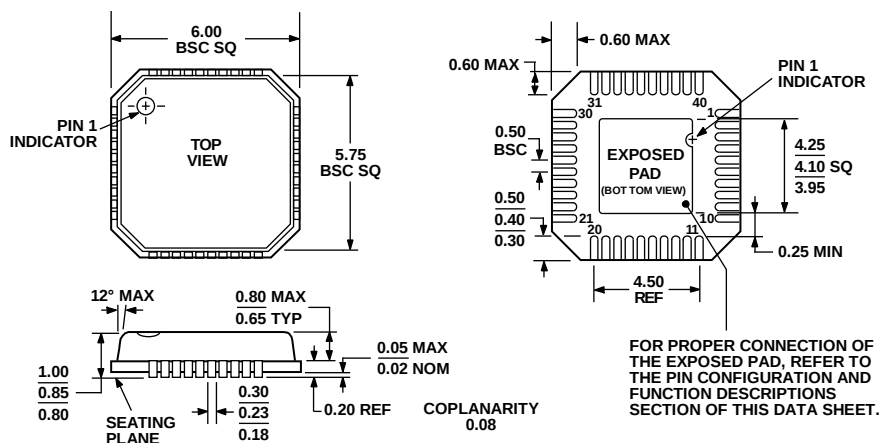
在线串行下载器

该串行下载器是一个Windows应用程序，用户可通过标准个人电脑(PC)的串口将已汇编程序串行下载到片内编程Flash/EE存储器。

所有开发系统都包含了基于UART的串行下载器，适用于订购指南中不含“I”后缀的ADuC7019/20/21/22/24/25/26/27/28/29器件。

还提供一种基于I²C的串行下载器和一个USB转I²C适配板(USB-EA-CONVZ)，参见www.analog.com。该基于I²C的串行下载器只适用于订购指南中含有“I”后缀的器件。

外形尺寸



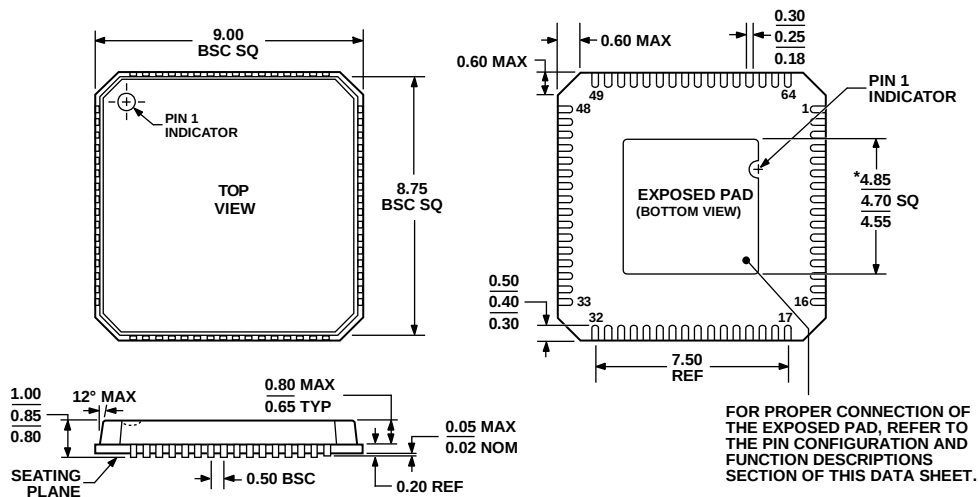
COMPLIANT TO JEDEC STANDARDS MO-220-VJJD-2

图84. 40引脚|脚架构芯片级封装[LFCSP_VQ]

6 mm × 6 mm, 超薄体

(CP-40-1)

尺寸单位: mm



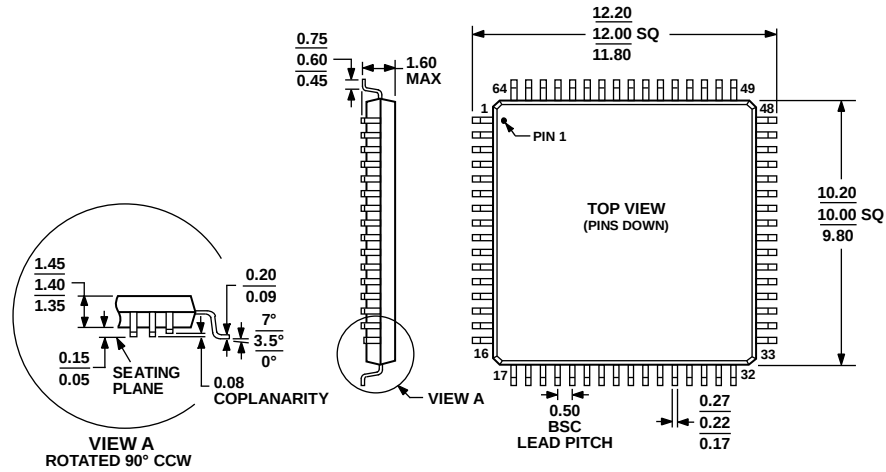
*COMPLIANT TO JEDEC STANDARDS MO-220-VMMD-4 EXCEPT FOR EXPOSED PAD DIMENSION

图85. 64引脚|脚架构芯片级封装[LFCSP_VQ]

9 mm × 9 mm, 超薄体

(CP-64-1)

尺寸单位: mm



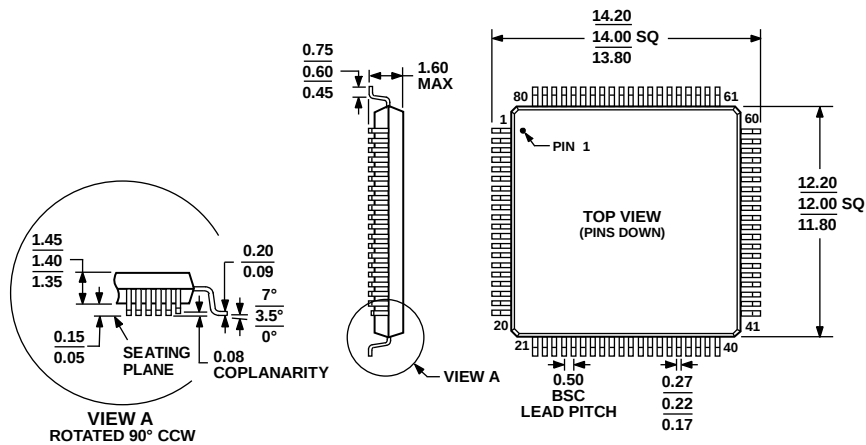
COMPLIANT TO JEDEC STANDARDS MS-026-BCD

图86. 64引脚薄型四方扁平封装[LQFP]

(ST-64-2)

尺寸单位: mm

051706-A



COMPLIANT TO JEDEC STANDARDS MS-026-BDD

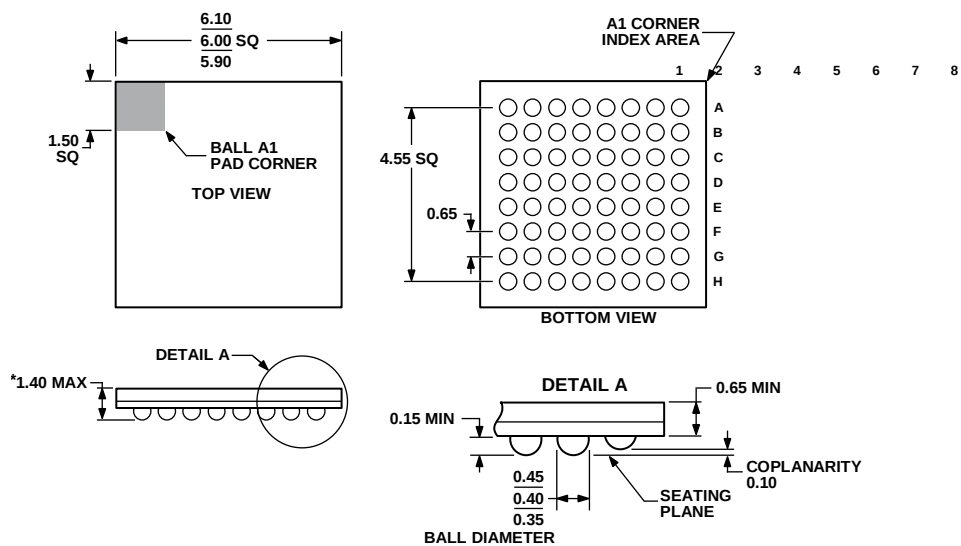
图87. 80引脚薄型四方扁平封装[LQFP]

(ST-80-1)

尺寸单位: mm

051706-A

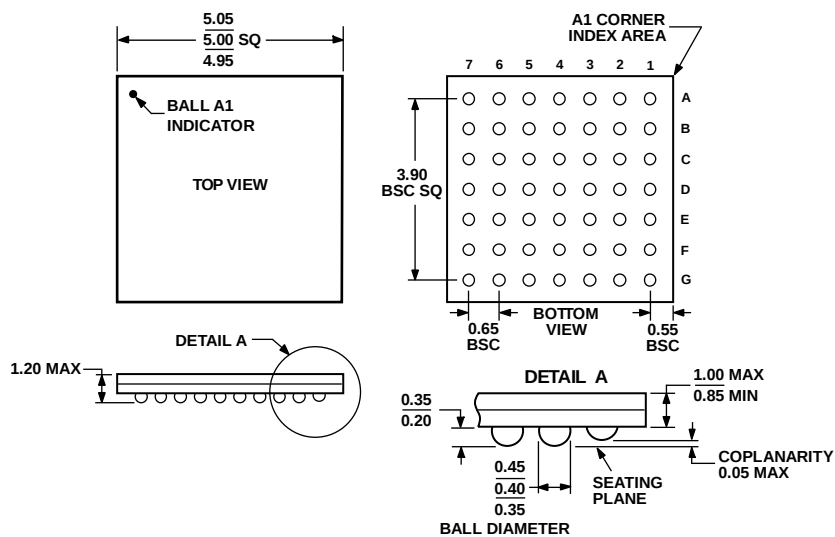
ADuC7019/20/21/22/24/25/26/27/28/29



039907-B

图88. 64引脚CSP_BGA封装
(BC-64-4)

尺寸单位: mm



012006-0

图89. 49引脚CSP_BGA封装
(BC-49-1)

尺寸单位: mm

订购指南

型号 ^{1,2}	ADC 通道 ³	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
ADuC7019BCPZ62I	5	3	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	2,500 750
ADuC7019BCPZ62I-RL	5	3	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7019BCPZ62IRL7	5	3	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7020BCPZ62	5	4	62 kB/8 kB	14	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	750
ADuC7020BCPZ62-RL7	5	4	62 kB/8 kB	14	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7020BCPZ62I	5	4	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7020BCPZ62I-RL	5	4	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	2,500
ADuC7020BCPZ62IRL7	5	4	62 kB/8 kB	14	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	750
ADuC7021BCPZ62	8	2	62 kB/8 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	2,500 750
ADuC7021BCPZ62-RL	8	2	62 kB/8 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7021BCPZ62-RL7	8	2	62 kB/8 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7021BCPZ62I	8	2	62 kB/8 kB	13	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	2,500
ADuC7021BCPZ62I-RL	8	2	62 kB/8 kB	13	I ² C	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7021BCPZ32	8	2	32 kB/4 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7021BCPZ32-RL7	8	2	32 kB/4 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	750
ADuC7022BCPZ62	10		62 kB/8 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	750
ADuC7022BCPZ62-RL7	10		62 kB/8 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7022BCPZ32	10		32 kB/4 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	
ADuC7022BCPZ32-RL	10		32 kB/4 kB	13	UART	−40°C至+125°C	40引脚 LFCSP_VQ	CP-40-1	2,500
ADuC7024BCPZ62	10	2	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	750
ADuC7024BCPZ62-RL7	10	2	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	
ADuC7024BCPZ62I	10	2	62 kB/8 kB	30	I ² C	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	
ADuC7024BCPZ62I-RL	10	2	62 kB/8 kB	30	I ² C	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	2,500
ADuC7024BSTZ62	10	2	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	ST-64-2	1,500
ADuC7024BSTZ62-RL	10	2	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	ST-64-2	
ADuC7025BCPZ62	12		62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	2,500
ADuC7025BCPZ62-RL	12		62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	
ADuC7025BCPZ32	12		32 kB/4 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	
ADuC7025BCPZ32-RL	12		32 kB/4 kB	30	UART	−40°C至+125°C	64引脚 LFCSP_VQ	CP-64-1	2,500
ADuC7025BSTZ62	12		62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LQFP	ST-64-2	1,000
ADuC7025BSTZ62-RL	12		62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 LQFP	ST-64-2	
ADuC7026BSTZ62	12	4	62 kB/8 kB	40	UART	−40°C至+125°C	80引脚 LQFP	ST-80-1	1,000
ADuC7026BSTZ62-RL	12	4	62 kB/8 kB	40	UART	−40°C至+125°C	80引脚 LQFP	ST-80-1	
ADuC7026BSTZ62I	12	4	62 kB/8 kB	40	I ² C	−40°C至+125°C	80引脚 LQFP	ST-80-1	
ADuC7026BSTZ62I-RL	12	4	62 kB/8 kB	40	I ² C	−40°C至+125°C	80引脚 LQFP	ST-80-1	1,000
ADuC7027BSTZ62	16		62 kB/8 kB	40	UART	−40°C至+125°C	80引脚 LQFP	ST-80-1	1,000
ADuC7027BSTZ62-RL	16		62 kB/8 kB	40	UART	−40°C至+125°C	80引脚 LQFP	ST-80-1	
ADuC7027BSTZ62I	16		62 kB/8 kB	40	I ² C	−40°C至+125°C	80引脚 LQFP	ST-80-1	
ADuC7027BSTZ62I-RL	16		62 kB/8 kB	40	I ² C	−40°C至+125°C	80引脚 LQFP	ST-80-1	1,000
ADuC7028BBCZ62	8	4	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 CSP_BGA	BC-64-4	2,500
ADuC7028BBCZ62-RL	8	4	62 kB/8 kB	30	UART	−40°C至+125°C	64引脚 CSP_BGA	BC-64-4	
ADuC7029BBCZ62	7	4	62 kB/8 kB	22	UART	−40°C至+125°C	49引脚 CSP_BGA	BC-49-1	4,000
ADuC7029BBCZ62-RL	7	4	62 kB/8 kB	22	UART	−40°C至+125°C	49引脚 CSP_BGA	BC-49-1	
ADuC7029BBCZ62I	7	4	62 kB/8 kB	22	I ² C	−40°C至+125°C	49引脚 CSP_BGA	BC-49-1	
ADuC7029BBCZ62I-RL	7	4	62 kB/8 kB	22	I ² C	−40°C至+125°C	49引脚 CSP_BGA	BC-49-1	4,000

ADuC7019/20/21/22/24/25/26/27/28/29

型号 ^{1, 2}	ADC 通道 ³	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
EVAL-ADuC7020MKZ							ADuC7020迷你套件		
EVAL-ADuC7020QSZ							ADuC7020 QuickStart 开发系统		
EVAL-ADuC7020QSPZ							ADuC7020 QuickStart 开发系统		
EVAL-ADuC7024QSZ							ADuC7024 QuickStart 开发系统		
EVAL-ADuC7026QSZ							ADuC7026 QuickStart 开发系统		
EVAL-ADuC7026QSPZ							ADuC7026 QuickStart Plus 开发系统		
EVAL-ADuC7028QSZ							ADuC7028 QuickStart 开发系统		

¹ Z = 符合RoHS标准的器件。
² ADuC7026和ADuC7027具有外部存储器接口。
³ ADuC7019的其中一个ADC通道具有内部缓冲。

¹ I²C指最初由Philips Semiconductors(现为NXP Semiconductors)开发的一种通信协议。

