

功能特性

- 工作电压范围: 2.8~5.5V
- 工作电流: 工作模式: 5mA/低功耗模式: 1uA
- 内置低噪声放大器, 1~128 倍可灵活配置
- 双信道差分信号输入
- 32 位高精度低零漂 Sigma-Delta ADC
- 采样率 6.25 Hz~51200 Hz
- 线性度 0.0005% FS, 24BIT 有效位数
- 噪声水平: $3\text{nV}/\sqrt{\text{Hz}}$ @ 128 倍 PGA
- 零漂: 小于 1uV @ 64/128 倍 PGA
- 集成 4.9MHz 内部高频 RC 时钟, -40~85°C

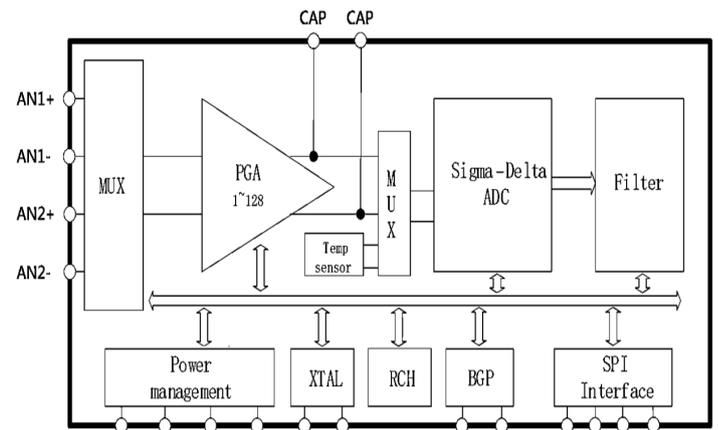
范围内温漂 1%

- 集成 10ppm/°C 的高精度基准参考电压源
- 内置温度传感器
- SPI 接口支持最高 10MHz
- 工作温度范围: -40~+85°C
- 存储温度范围: -40~+125°C
- 封装样式: SSOP20

应用场合

- 工业程控
- 电子秤
- 液体/气体化学分析
- 血液计
- 智能变换器

功能方块图



概述

CS5552 为一款 SPI 接口的双信道、32 位高精度 ADC 芯片, 内置 1~128 倍可程序设计的低噪声仪表放大器、高精度 Sigma-Delta ADC, 同时内部集成温度传感器、高精度基准电压源、晶体起振电路、高频内部 RC 时钟源。

ADC 实际有效精度 (ENOB) 24.3BIT@1 倍 PGA, 22.9BIT@64 倍 PGA, 等效输入噪声低至 $3\text{nV}/\sqrt{\text{Hz}}$, 零漂 0.5uV, 零漂温度系数低于 $5\text{nV}/\text{°C}$ 。输出码率可配置为 6.25Hz 至 51200Hz。

可用于各类电子秤、分析天平、工业过程控制、直流/交流电能测量、耳温枪等需要高精度、低零漂的应用场合。

目录 TABLE OF CONTENTS

功能特性	1	5	寄存器描述	16
功能方块图	1	5.1	寄存器地址	16
应用场合	1	5.2	OS_CHx/GAIN_CHx 寄存器	17
概述	1	5.3	CONV_CONFx 寄存器	18
修订追踪	2	5.4	SYS_CONFx 寄存器	19
1 电器规格特性	3	5.4.1	SYS_CONF0	20
1.1 极限参数	3	5.4.2	SYS_CONF1	21
1.2 工作参数	3	5.4.3	SYS_CONF2	23
1.3 ESD/LU 性能	3	5.5	D_TARG 寄存器	24
1.4 GPIO 参数	3	5.6	CONV_DATA 寄存器	24
1.5 ADC 性能指针	4	6	ADC 校准	25
2 芯片引脚	6	6.1	校准概述	25
2.1 引脚定义	6	6.2	Offset 自校准	25
3 功能模块描述	7	6.3	Offset 系统校准	25
3.1 电源管理模块	7	6.4	Gain 系统校准	26
3.2 晶体起振模块	7	6.5	正常转换时的数据校准	26
3.3 高频 RCH 时钟	8	7	温度传感器	26
3.4 BGP 电路	8	8	ADC 测试图表	27
3.5 MUX 信号选择电路	8	8.1	不同增益设置下的噪声值	27
3.6 PGA 电路	8	8.2	不同数据码率设置下的噪声值	29
3.7 温度传感器	8	8.3	增益和 Offset 的温漂	30
3.8 Sigma-Delta ADC	9	8.4	电源抑制比(PSRR)	31
3.9 数字滤波器	9	8.5	共模抑制比(CMRR)	32
3.10 功耗模式	9	9	应用参考线路	33
3.11 SPI 界面	9	10	PACKAGING 封装尺寸图	34
4 SPI 接口协议	9	11	应用开发笔记	35
4.1 SPI 和校验	9	12	参考 DEMO 程序	37
4.2 读写命令帧	10			
4.3 转换命令帧	12			
4.4 SPI 界面复位	15			
4.5 SPI 读写时序参数	15			

修订追踪

2020/3/20 初稿

2020/9/25 修订 参考应用电路

2020/9/25 新增 程序开发笔记

2020/9/25 新增 程序 DEMO CODE

1 电器规格特性

1.1 极限参数

当外部输入或是环境参数超过下面条件时，很可能会对于芯片造成损坏或是缩短其使用寿命。下表只代表会造成损坏的范围，不代表可以正常工作的范围。

Table 1-1 极限参数表

Symbol	Ratings	Min	Max	Unit
AV+/DV+	电源电压	-0.3	+6	V
Vsig	信号输入信号	-0.3	+6	V
TS	存储温度	-50	+150	°C
TJ	Junction Temperature under bias	-40	+125	°C

1.2 工作参数

Table 1-2 工作参数表

Symbol	Parameter	Min	Typ	Max	Unit
AV+/DV+	IO 口电压	2.8	5	5.5	V
I _{ACTIVE}	工作电流		5		mA
I _{PD}	休眠电流		0.6		uA
VPOR	上电复位电压	1.9	2	2.1	
VLVD	掉电监测电压	2.8	2.9	3	
TA	温度范围	-40	25	85	°C

1.3 ESD/LU 性能

Table 1-3 ESD/Latch-Up 性能指针

Symbol	Parameter	Min	Max	Unit
ESD(HBM)	HBM 模型的 ESD 放电电压	-4000	4000	V
Latch-Up	Latch-Up 测试电流 (@85°C)	-200	200	mA

1.4 GPIO 参数

Table 1-4 GPIO 参数表

Symbol	Parameter	DV+	Min	Typ	Max	Unit
VIH	输入信号高阈值	5V	4		5.5	V
VIL	输入信号低阈值	5V	-0.3		1	V
VT+	施密特由低变高电压的阈值	5V	2.72	2.92	3.17	V
VT-	施密特由高变低电压的阈值	5V	1.85	2	2.17	V
IIH	输入高电平的电流	5V			+1	uA
IIL	输入低电平的电流	5V	-1			uA
VOL	输出低电平 (@IOL 电流条件)	5V			0.4	V
VOH	输出高电平 (@IOH 电流条件)	5V	4			V
IOL	输出低电平电流@VOL (max)	5V	4.9	8.8	13.9	mA
IOH	输出高电平电流@VOH (min)	5V	5.5	15.6	29.9	mA

1.5 ADC 性能指针

Table 1-5 ADC 性能指针表

AV+=5V DV+=5V VREF+=5V 条件条件下测试

Parameter	Min	Typ	Max	Unit
精度				
线性度(Linearity)		±0.0005	±0.001	%FS
有效位数 (ENOB)		24.3@PGA=1 22.9@PGA=64		BIT
无噪声位数 (Noise Free Bits)		21.8@PGA=1 20.4@PGA=64		BIT
等效噪声密度 (Noise Floor)		3.1	4.5	nV/ \sqrt{Hz}
零漂 (Offset)		0.5	1	uV
零漂温漂 (Offset drift)		5		nV/°C
增益误差(Gain error)		0.1		%
增益温漂 (Gain drift)		1		ppm/°C
信号输入				
输入信号共模范围	AV-		AV+	V
差分输入电流		2		nA
信号输入阻抗		>1G		Ω
输入共模抑制比 (CMRR)		140		dB
基准电压				
VREF+-VREF-	1		AV+-AV-	V
差分输入电流		2		nA
内置基准电压	1.17	1.2	1.23	V
内置基准电压温度系数		10		ppm/°C
时钟				
ADC 转换速率 (Data Rate)	6.25		51200	Hz
晶体时钟频率		4.9152		MHz
内部 RC 时钟频率		4.9		MHz
电源				
AV+电源范围	2.8	5	5.5	V
DV+电源范围	2.8	5	5.5	V
电源抑制比 (PSRR)		140		dB

Table 1-6 等效输入 RMS 噪声(nV)

AV+=5V DV+=5V VREF+=5V 条件条件下测试

数据码率 (Hz)	-3dB 带宽 (Hz)	噪声(nV)@PGA(倍)							
		128	64	32	16	8	4	2	1
6.25	1.64	7.95	9.91	13.81	21.63	37.25	68.50	131.00	250.00
12.5	3.27	11.25	14.01	19.53	30.58	52.68	96.87	185.26	353.55
25	6.54	15.91	19.81	27.63	43.25	74.50	137.00	262.00	500.00
50	13.09	22.49	28.02	39.07	61.16	105.36	193.75	370.52	707.11
100	26.18	31.81	39.63	55.25	86.50	149.00	274.00	524.00	1000.00
200	52.36	44.99	56.04	78.14	122.33	210.72	387.49	741.05	1414.21
400	104.71	63.63	79.25	110.50	173.00	298.00	548.00	1048.00	2000.00
800	209.42	89.98	112.08	156.27	244.66	421.44	774.99	1482.10	2828.43
1600	418.85	127.25	158.50	221.00	346.00	596.00	1096.00	2096.00	4000.00
3200	837.70	179.96	224.15	312.54	489.32	842.87	1549.98	2964.19	5656.85
6400	1306.12	254.50	317.00	442.00	692.00	1192.00	2192.00	4192.00	8000.00
12800	2612.24	359.92	448.31	625.08	978.64	1685.74	3099.96	5928.38	11313.71
25600	5224.49	509.00	634.00	884.00	1384.00	2384.00	4384.00	8384.00	16384.00
51200	10448.98	1073.39	1603.72	2664.38	4785.70	9028.34	17513.62	34484.18	68425.31

Table 1-7 ENOB

AV+=5V DV+=5V VREF+=5V 条件条件下测试

数据码率 (Hz)	-3dB 带宽 (Hz)	噪声(nV)@PGA(倍)							
		128	64	32	16	8	4	2	1
6.25	1.64	22.2	22.9	23.4	23.8	24.0	24.1	24.2	24.3
12.5	3.27	21.7	22.4	22.9	23.3	23.5	23.6	23.7	23.8
25	6.54	21.2	21.9	22.4	22.8	23.0	23.1	23.2	23.3
50	13.09	20.7	21.4	21.9	22.3	22.5	22.6	22.7	22.8
100	26.18	20.2	20.9	21.4	21.8	22.0	22.1	22.2	22.3
200	52.36	19.7	20.4	20.9	21.3	21.5	21.6	21.7	21.8
400	104.71	19.2	19.9	20.4	20.8	21.0	21.1	21.2	21.3
800	209.42	18.7	19.4	19.9	20.3	20.5	20.6	20.7	20.8
1600	418.85	18.2	18.9	19.4	19.8	20.0	20.1	20.2	20.3
3200	837.70	17.7	18.4	18.9	19.3	19.5	19.6	19.7	19.8
6400	1306.12	17.2	17.9	18.4	18.8	19.0	19.1	19.2	19.3
12800	2612.24	16.7	17.4	17.9	18.3	18.5	18.6	18.7	18.8
25600	5224.49	16.2	16.9	17.4	17.8	18.0	18.1	18.2	18.2
51200	10448.98	15.2	15.6	15.8	16.0	16.1	16.1	16.1	16.2

Table 1-8 Noise Free Bits

AV+=5V DV+=5V VREF+=5V 条件条件下测试

数据码率 (Hz)	-3dB 带宽 (Hz)	噪声(nV)@PGA(倍)							
		128	64	32	16	8	4	2	1
6.25	1.64	19.7	20.4	20.9	21.3	21.5	21.6	21.7	21.8
12.5	3.27	19.2	19.9	20.4	20.8	21.0	21.1	21.2	21.3
25	6.54	18.7	19.4	19.9	20.3	20.5	20.6	20.7	20.8
50	13.09	18.2	18.9	19.4	19.8	20.0	20.1	20.2	20.3
100	26.18	17.7	18.4	18.9	19.3	19.5	19.6	19.7	19.8
200	52.36	17.2	17.9	18.4	18.8	19.0	19.1	19.2	19.3
400	104.71	16.7	17.4	17.9	18.3	18.5	18.6	18.7	18.8
800	209.42	16.2	16.9	17.4	17.8	18.0	18.1	18.2	18.3
1600	418.85	15.7	16.4	16.9	17.3	17.5	17.6	17.7	17.8
3200	837.70	15.2	15.9	16.4	16.8	17.0	17.1	17.2	17.3
6400	1306.12	14.7	15.4	15.9	16.3	16.5	16.6	16.7	16.8
12800	2612.24	14.2	14.9	15.4	15.8	16.0	16.1	16.2	16.3
25600	5224.49	13.7	14.4	14.9	15.3	15.5	15.6	15.7	15.7
51200	10448.98	12.7	13.1	13.3	13.5	13.6	13.6	13.6	13.7

2 芯片引脚

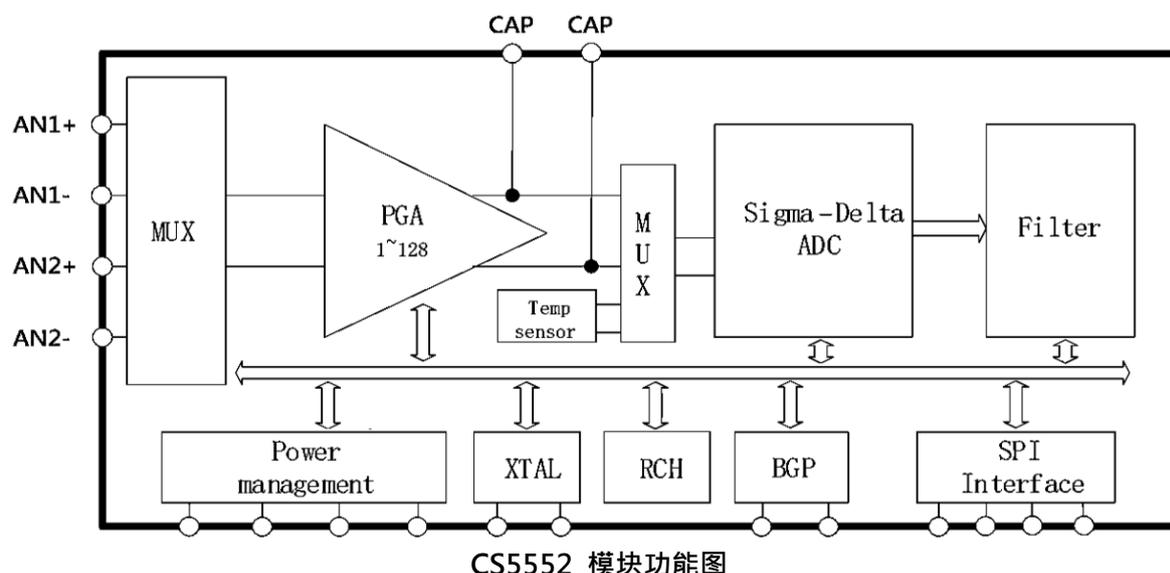
2.1 引脚定义

DIFFERENTIAL ANALOG INPUT	AIN1+	1	20	AIN2+	DIFFERENTIAL ANALOG INPUT
DIFFERENTIAL ANALOG INPUT	AIN1-	2	19	AIN2-	DIFFERENTIAL ANALOG INPUT
AMPLIFIER CAPACITOR CONNECT	CAP	3	18	VREF+	VOLTAGE REFERENCE INPUT
AMPLIFIER CAPACITOR CONNECT	CAP	4	17	VREF-	VOLTAGE REFERENCE INPUT
POSITIVE ANALOG POWER	VA+	5	16	DGND	DIGITAL GROUND
NEGATIVE ANALOG POWER	VA-	6	15	VD+	POSITIVE DIGITAL POWER
NOT CONNECT	NC	7	14	CS	CHIP SELECT
NOT CONNECT	NC	8	13	SDI	SERIAL DATA INPUT
MASTER CLOCK	OSC2	9	12	SDO	SERIAL DATA OUT
MASTER CLOCK	OSC1	10	11	SCLK	SERIAL CLOCK INPUT

Table 2-1

序号	引脚名称	输入/输出	说明
Pin Number	名称	类型	描述
1	VIN1+	输入	信号信道 0 正输入端
2	VIN1-	输入	信号信道 0 负输入端
3	CAP	输入/输出	去耦电容埠 CAP，在 CAP 之间接一个 100nF 的瓷片电容。
4	CAP	输入/输出	如设置的输出数据码率在 3.2kHz 以上，则应改为 1nF 或 510pF 的电容
5	VA+	电源	模拟电源，AV+ 和 AV- 之间接一个大于等于 1uF 的瓷片电容模拟地
6	VA-	地	模拟地
7	NC	输出	空 PIN 不须外接
8	NC	输出	空 PIN 不须外接
9	OSC1	输入	晶体输入，在 XIN/XOUT 之间接一个 4.9152MHz 的晶体，无需外部电容
10	OSC2	输出	晶体输出
11	SCLK	输入	SPI 时钟输入，建议片外接 10K Ohm 上拉至电源的电阻
12	SDO	输出	SPI 数据输出，建议片外接 10K Ohm 上拉至电源的电阻
13	SDI	输入	SPI 数据输入，建议片外接 10K Ohm 上拉至电源的电阻
14	CS	输入	SPI 片选输入，低电平有效
15	VD+	电源	数字电源，DV+ 和 DGND 之间接一个大于 0.1uF 的瓷片电容
16	DGND	地	数字地
17	VREF-	输入/输出	基准电压源负输入端，一般情况接 AV-
18	VREF+	输入/输出	基准电压源正输入端，VREF+ 和 VREF- 之间接一个大于等于 1uF 的瓷片电容
19	VIN2-	输入	信号信道 1 负输入端
20	VIN2+	输入	信号信道 1 正输入端

3 功能模块描述



如上图所示，芯片内部包含电源管理模块、晶体起振模块、高频 RCH 时钟、内部基准源 (BGP) 模块、输入信号选择电路(MUX)、可程序设计增益放大器 (PGA)、Sigma-Delta ADC 模块、数字滤波器模块、SPI 接口模块。

3.1 电源管理模块

芯片由 DV+/DGND/AV+/AV-两组电源供电，正常工作的供电范围为 2.8~5.5V。电源模块内置上电复位 (POR) 电路，在上电之初将为芯片提供复位信号。

电源模块内还集成了欠压监测电路，当监测到 AV+/DV+电源低于 3V 时，SYS_CONF0 寄存器里的 PW_LV 将置 1。

3.2 晶体起振模块

芯片内部集成晶体起振模块，只需在 XIN/XOUT 引脚上接入一个晶体即可起振，晶体频率范围 2MHz~8Mhz，默认选用 4.9152MHz。

3.3 高频 RCH 时钟

芯片内部集成 4.9MHz 的高频 RCH 时钟，该时钟作为晶体时钟的备份，如果晶体时钟因意外停止工作，在停振 200ms 之后，系统将自动切换到 RCH 时钟上。

芯片内部 RCH 时钟在 -40~85 度范围内随温度的频率变化小于 1%，但芯片之间的 RCH 频率存在固有偏差。如系统应用在无晶体的场合，且希望 RCH 频率精度较高时，需由 MCU 做额外校正，校正值配置到 SYS_CONF2 寄存器里的 RCTRIM<3:0>，详见 SYS_CONFx 寄存器章节。有晶体的应用场合则无需配置此寄存器。

3.4 BGP 电路

芯片内置高精度带隙基准 (BGP) 电路，产生 1.2V 的基准电压。该基准电压温度系数的典型值为 10ppm/°C。

在桥式传感器等应用下，采用外部输入的基准电压。外部基准从 VREF+ 和 VREF- 引脚接入，SYS_CONF1 寄存器里的 VRS 采用默认的 '0'，此时内部产生的基准电压不起作用。

在无外部基准电压的应用下，需将 SYS_CONF1 寄存器里的 VRS 配置为 '1'，则内部基准电压从 VREF+ 和 VREF- 送出，且 VREF+ 和 VREF- 之间需加 1 个 1uF 去耦电容。

基准电压的温度系数可通过设置 SYS_CONF2 寄存器里的 REFTRIM<2:0> 进行微调。详见 SYS_CONFx 寄存器章节。

3.5 MUX 信号选择电路

芯片有两路输入信号引脚，以及一路内部温度传感器信号，ADC 可通过时分复用的方式对这 3 路信号进行采样。

两路外部信号的选择是通过 CONV_CONFx 寄存器里的 CHS 进行设置，CHS=0 时选择 VIP0/VIN0 引脚的信号进行转换，CHS=1 时选择 VIP1/VIN1 引脚。但如果 CONV_CONFx 寄存器里的 TMPEN 设置为 1，则选择对内部温度传感器的信号进行转换。详见 CONV_CONFx 寄存器章节。

3.6 PGA 电路

芯片内部集成一个 1~128 倍可程序设计的高精度仪表放大器，增益可通过 CONV_CONFx 寄存器里的 GA<2:0> 进行设置。详见 CONV_CONFx 寄存器章节。

3.7 温度传感器

芯片内部集成 2°C 精度的温度传感器，传感器计算公式详见温度传感器章节。

3.8 Sigma-Delta ADC

芯片内集成一路高性能的 Sigma-Delta ADC (性能指针见 ADC 性能指针章节), ADC 转换产生的高频量化码流送给后续的数字滤波器 (DSP) 电路进行处理, 并最终得到 32BIT ADC 数据。在 50Hz 工作模式下 (由 SYS_CONF1 寄存器的 FR_SEL 控制), ADC 默认的工作频率为 819.2kHz。也可以通过配置 SYS_CONF2 寄存器的 ADCKDIV2 将其配置为 409.6kHz。详见 SYS_CONFx 寄存器章节。

3.9 数字滤波器

数字滤波器对 ADC 转换产生的高频量化码流进行处理, 最终获得 32BIT ADC 数据。滤波器输出频率可通过 CONV_CONFx 寄存器里的 DR<3:0> 配置为 6.25~51200Hz。详见 CONV_CONFx 寄存器章节。

3.10 功耗模式

芯片可通过设置 SYS_CONF1 寄存器的 POWD=1, 使芯片进入低功耗模式。此模式下 PGA/ADC/晶体起振电路/BGP/RCH 等模块都会关闭, 功耗小于 1uA。

3.11 SPI 界面

SPI 接口部分详见 SPI 接口协议章节

4. SPI 接口协议

SPI 接口分为两种命令帧, 一种是用来读写寄存器的读写命令帧, 另外一种是用来启动 ADC 转换的转换命令帧, 两种命令帧使用第一个字节的第一个 Bit 来区分, 若为 0, 则为读写命令帧, 若为 1, 则为转换命令帧。

4.1. SPI 和校验

当配置寄存器 SYS_CONF0 的 CKS_EN 位被设置为 1 后, SPI 接口会进入和校验模式, 包含读写命令帧里的数据与转换命令帧得到的 ADC 数据, 都会包含和校验帧。和校验帧针对写入或者读出的数据做保护, 每个 32bits 数据会被拆成四个字节, 将四个字节总合起来再加上 0x5A 后, 取最低 8 bit 即为和校验帧, 如以下公式和校验 = Data Byte 0 + Data Byte 1 + Data Byte 2 + Data Byte 3 + 0x5A 在每个 32bits 数据后端, 都需要加上和校验帧, 若和校验不符合, 则该命令会被忽略 (写命令帧), 或是该数据为无效数据 (读命令帧或是转换命令帧得到的 ADC 数据)。开启和校验模式后的时序图见下章节。

4.2. 读写命令帧

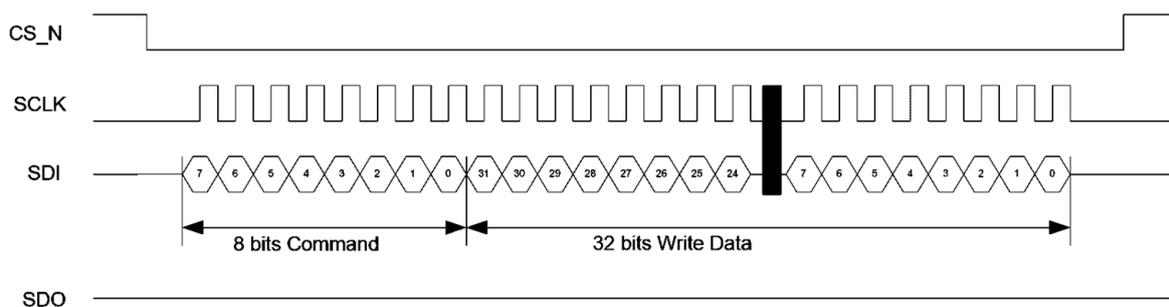
读写命令帧的第一帧为命令帧，格式如下

Table 4-1 读写命令帧结构

Bit							
7	6	5	4	3	2	1	0
0	ADDR				R/W	ARRAY	PC

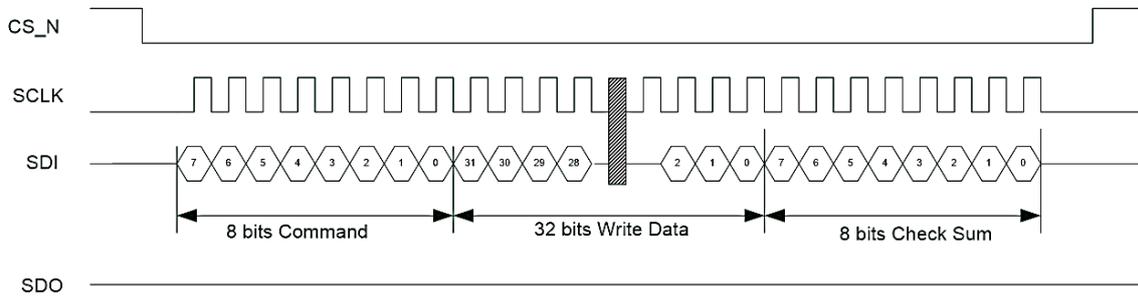
BIT	名称	描述
7	起始位	必须为 0
6:3	ADDR	读写命令帧的目标寄存器地址，具体说明详见寄存器描述章节： 0x0: OS_CH0 0x1: GAIN_CH0 0x2: OS_CH1 0x3: GAIN_CH1 0x4: CONV_CONF0 0x5: CONV_CONF1 0x6: SYS_CONF0 0x7: SYS_CONF1 0x8: SYS_CONF2 0x9: D_TARG 0xA: CONV_DATA 其他: 不可擦写
2	R/W	读写类型选择 0: 写 1: 读
1	ARRAY	单一或是连续寄存器读写 0: 单一寄存器读写 1: 连续寄存器读写，此时会固定由 ADDR=0 寄存器开始读写，到 ADDR=9 结束读写，ADDR 字段会被忽略。
0	PC	Bit 7 到 Bit 1 之奇偶校验位，当 Bit 7~Bit 1 有奇数个 1 时，PC 应为 1，当 B7~B1 有偶数个 1 时，PC 应为 0。若此奇偶校验位错误，则该命令不被执行，且 SYS_CONF0 中的 ERR_CKS 位置会被置 1。

下图为单一寄存器写命令帧不包含和校验的时序



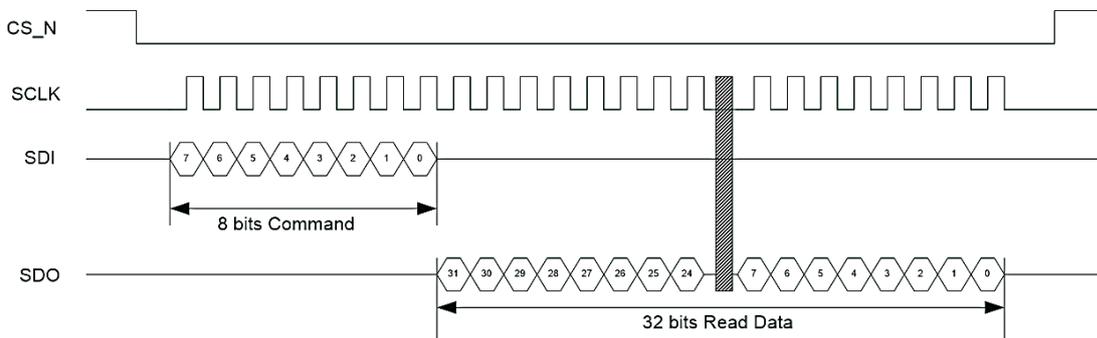
SPI 单一寄存器写命令帧时序 (不包含和校验)

下图为单一寄存器写命令帧包含和校验的时序



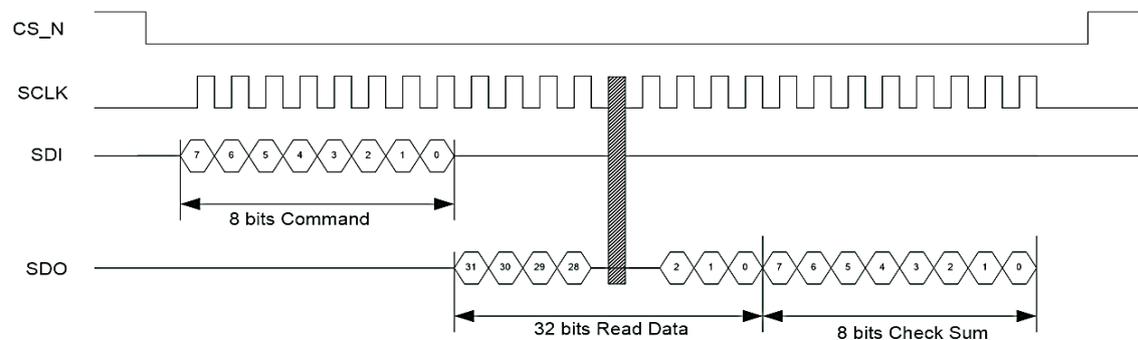
SPI 单一寄存器写命令帧时序 (包含和校验)

下图为单一寄存器读命令帧不包含和校验的时序



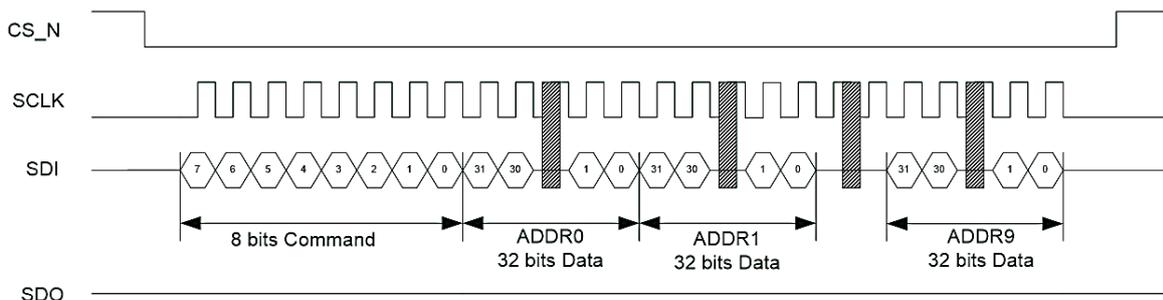
SPI 单一寄存器读命令帧时序 (不包含和校验)

下图为单一寄存器读命令帧包含和校验的时序



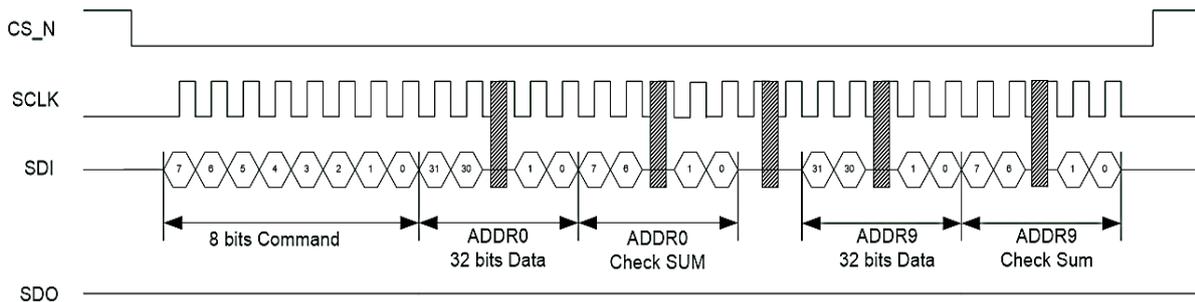
SPI 单一寄存器读命令帧时序 (包含和校验)

下图为连续寄存器写命令帧不包含和校验的时序



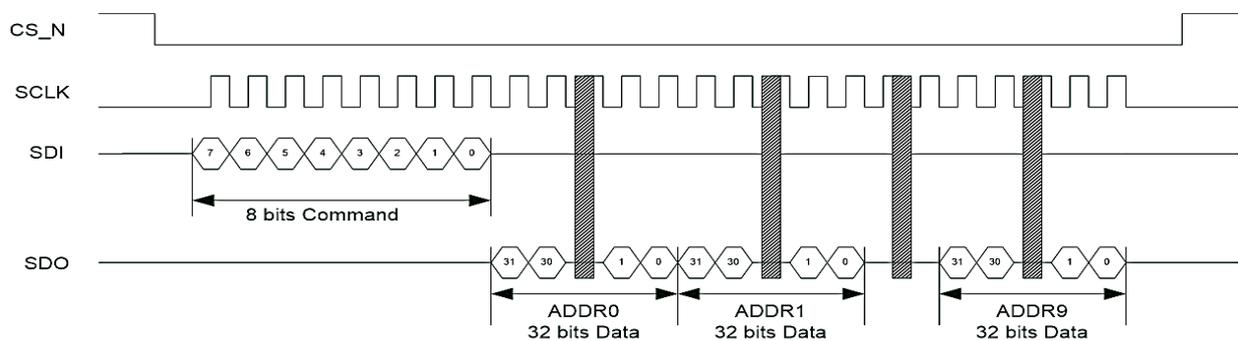
SPI 连续寄存器写命令帧时序 (不包含和校验)

下图为连续寄存器写命令帧包含和校验的时序



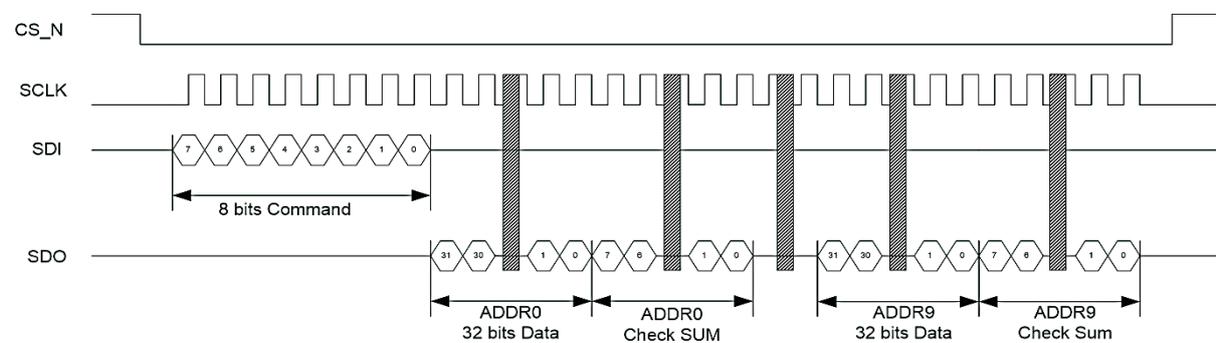
SPI 连续寄存器写命令帧时序 (包含和校验)

下图为连续寄存器读命令帧不包含和校验的时序



SPI 连续寄存器读命令帧时序 (不包含和校验)

下图为连续寄存器读命令帧包含和校验的时序



SPI 连续寄存器读命令帧时序 (包含和校验)

4.3. 转换命令帧

转换命令帧的第一帧为命令帧，格式如下

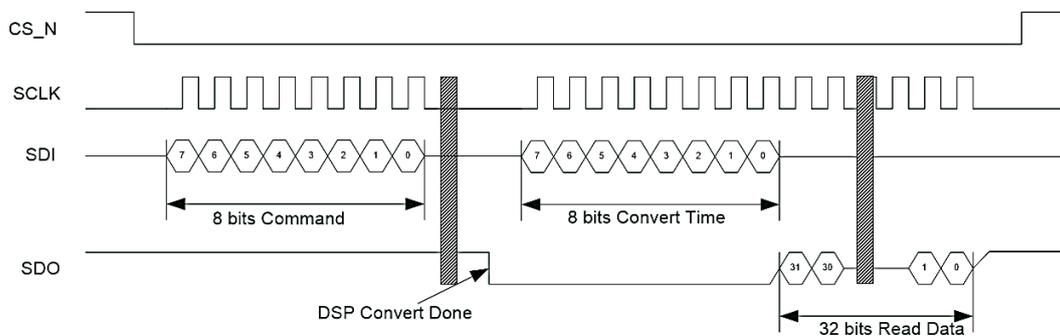
Table 4-2 转换命令帧结构

Bit							
7	6	5	4	3	2	1	0
1	0	CONV_CONF		CONV_MOD		PC	

Bit	名称	描述
7	起始位	必须为 1
6	保留位	必须为 0
5:4	CONV_CONF	转换配置选择，具体说明详见 CONV_CONFx 寄存器 0x0: 使用 CONV_CONF00 作为转换设置参数寄存器 0x1: 使用 CONV_CONF01 作为转换设置参数寄存器 0x2: 使用 CONV_CONF10 作为转换设置参数寄存器 0x3: 使用 CONV_CONF11 作为转换设置参数寄存器
3:1	CONV_MOD	转换模式选择 0x0: 正常单次转换模式 0x1: 正常连续转换模式 0x2: Offset 自校准模式 0x5: Offset 系统校准模式 0x6: Gain 系统校准模式 其他: 保留
0	PC	Bit 7 到 Bit 1 之奇偶校验位，当 Bit 7~Bit 1 有奇数个 1 时，PC 应为 1，当 B7~B1 有偶数个 1 时，PC 应为 0。若此奇偶校验位错误，则该命令不被执行，且 SYS_CONF0 中的 ERR_CKS 位置会被置 1。

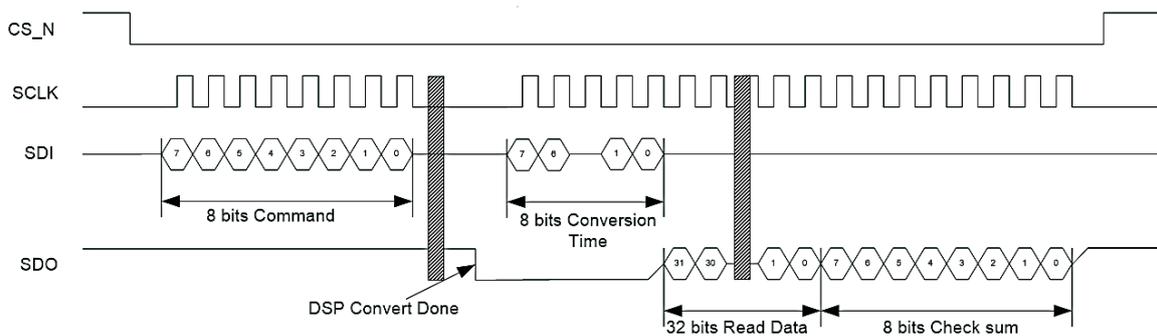
转换命令帧时序与读时序主要差别在于转换命令发出后，需要等待芯片内 ADC (DSP) 转换完成。在转换完成前，SDO 引脚是高阻抗状态，需靠片外上拉电阻拉高到 DV+电平。当 ADC 转换完成后，SDO 引脚会输出低电平，此时主控 MCU 需打 8bits 的转换时钟，同时 SDI 上的数据为 0xA5 之外的其他值，然后可以开始进行 32bits 转换数据读取。

下图为单次转换命令帧不包含和校验的时序



单次转换命令帧时序 (不包含和校验)

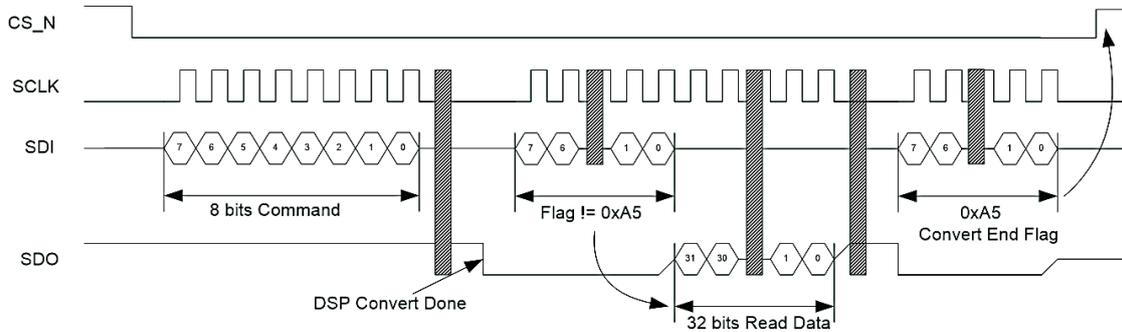
下图为单次转换命令帧包含和校验的时序



SPI 单次转换命令帧时序 (包含和校验)

若选择的是连续转换模式，当 32bits 转换数据传完之后，SDO 又会变为高阻模式，直到下次 ADC 转换完成，将再把 SDO 拉到 0。若主控 MCU 想要停止连续转换模式，则必须在 SDO 变低之后的 8bits 转换时钟时，在 SDI 上送入 0xA5，则在本次数据传递完成后，芯片会回到待命状态。

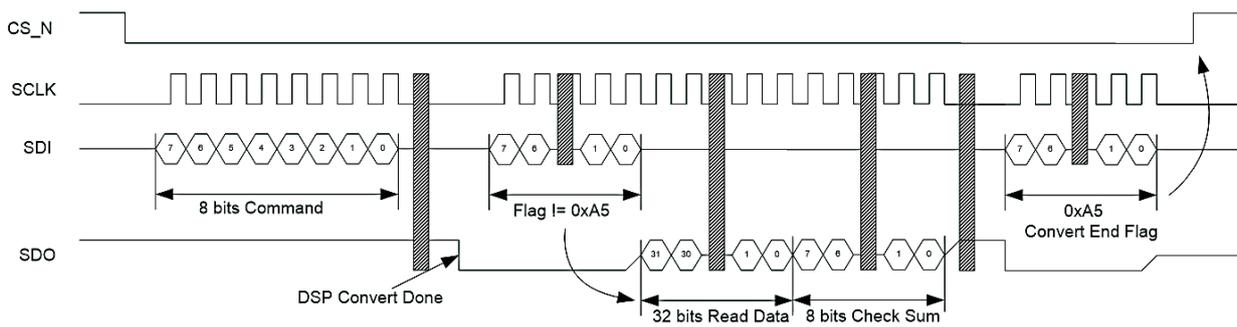
下图为不包含和校验、且连续转换期间 CS_N 固定低时的时序图。



SPI 连续转换命令帧时序 (不包含和校验,CS_N 固定低)

下图为包含和校验、且连续转换期间 CS_N 固定低时的时序图。

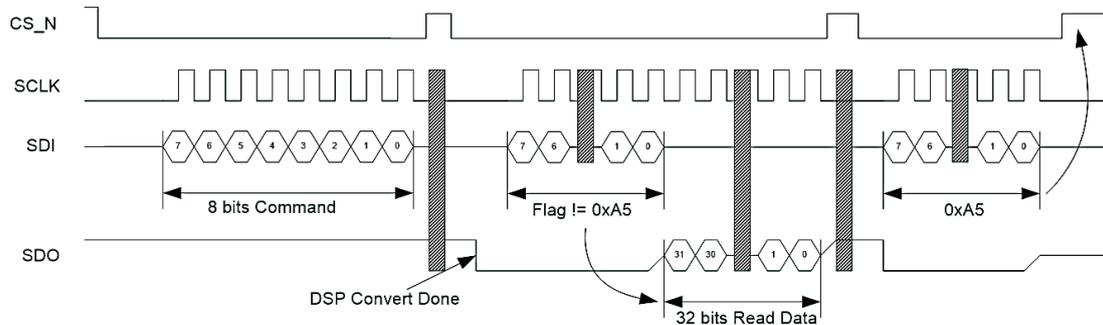
开启和校验的模式下，如果转换数据的和校验连续出错，建议对 SPI 接口进行强制复位（见 SPI 界面复位章节），并重新配置开启转换。



SPI 连续转换命令帧时序 (包含和校验,CS_N 固定低)

下图为不包含和校验、且连续转换时 CS_N 可周期性置 1 的时序图（进入此模式前需将 SYS_CONF0 寄存器里的 CSHIGH_MODE 置 1）。

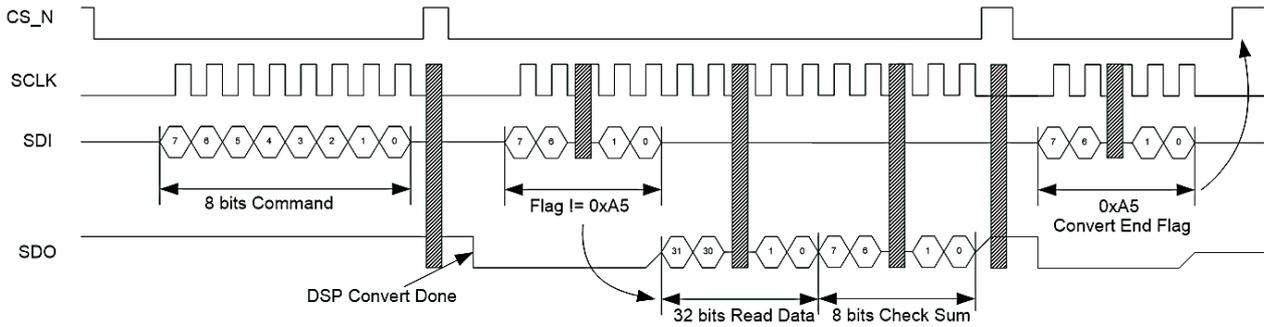
CS_N 置‘1’只可在当次数据读取完成之后发生，这期间 SPI 接口可复用来读取另一颗芯片的数据。另一颗芯片资料读完后，再将当前芯片的 CS_N 置‘0’，如下一个数据完成转换，则 SDO 将置‘0’，通知主控 MCU 取资料。



SPI 连续转换命令帧时序 (不包含和校验,CS_N 可变)

下图为包含和校验、且连续转换时 CS_N 可周期性置' 1' 的时序图(进入此模式前需将 SYS_CONF0 寄存器里的 CSHIGH_MODE 置 1)。

开启和校验的模式下，如果转换数据的和校验连续出错，建议对 SPI 接口进行强制复位（见 SPI 界面复位章节），并重新配置开启转换。



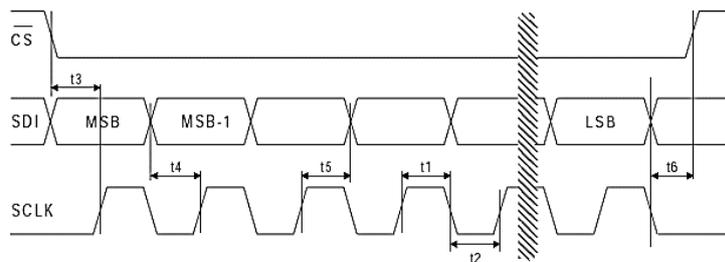
SPI 连续转换命令帧时序 (包含和校验,CS_N 可变)

4.4. SPI 界面复位

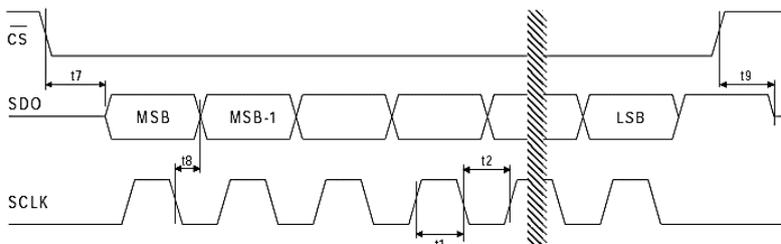
芯片上电完成后，内部 POR 会将 SPI 接口复位，但仍建议在 SDI 上打入 Byte0=0x00，Byte1=0xA5，Byte2=0xFF，Byte3=0x5A，相应的 SCLK 上连续 32 个时钟信号，以强制 SPI 接口复位。

0x00A5FF5A 指令在任何时候都可以复位 SPI 接口，复位完成后等待 1 us 后可以重新开始操作 SPI 指令。

4.5. SPI 读写时序参数



SPI 写时序参数图



SPI 读时序参数图

Table 4-3 SPI 读写时序参数

	标识	最小	典型	最大	单位
SPI 时序					
SPI 时钟频率	SCLK	0		10	MHz
SPI 时钟脉宽	t1(高)	50			ns
	t2(低)	50			ns
SDI 写时序					
CS_N 片选到第一个时钟延时	t3	10			ns
DATA 领先时钟上升沿的建立时间	t4	10			ns
DATA 在时钟沿后的稳定时间	t5	20			ns
时钟下降沿后到 CS_N 上升的延时	t6	20			ns
SDI 读时序					
CS_N 信号变低到有效数据	t7			30	ns
SCLK 下降沿到新数据输出延时	t8			30	ns
CS_N 信号变高到 SDO 进高阻态的延时	t9			30	ns

SPI 写入数据的时候，SDI 的数据在 SCLK 的下降沿变化。

SPI 读出数据的时候，SDO 的数据变化在 SCLK 下降沿之后变化。

5. 寄存器描述

5.1. 寄存器地址

CS5552 内共有 10 个 32bits 寄存器，可以通过读写命令帧的 ADDR 未来选择要读取或是写入的寄存器地址，下表列出各个寄存器对应的地址与初始值

Table 5-1 寄存器地址表

地址	名称	类型	描述	初始值
0x0	OS_CH0	R/W	ADC 通道 0 Offset 设置	0x00000000
0x1	GAIN_CH0	R/W	ADC 通道 0 Gain 设置	0x02000000
0x2	OS_CH1	R/W	ADC 通道 1 Offset 设置	0x00000000
0x3	GAIN_CH1	R/W	ADC 通道 1 Gain 设置	0x02000000
0x4	CONV_CONF0	R/W	CONV_CONF00 与 CONV_CONF01 设置寄存器	0x00000000
0x5	CONV_CONF1	R/W	CONV_CONF10 与 CONV_CONF11 设置寄存器	0x00000000
0x6	SYS_CONF0	R/W	系统设置寄存器 0	0x00000001
0x7	SYS_CONF1	R/W	系统设置寄存器 1	0x00000000
0x8	SYS_CONF2	R/W	系统设置寄存器 2	0x00000000
0x9	D_TARG	R/W	增益校准目标寄存器	0x3FFFFFFF
0xA	CONV_DATA	R	转换数据寄存器	--
其他	保留			--

下面章节针对各个寄存器做详细介绍，需留意各寄存器中的“保留”位，需保持其默认值。

5.2.OS_CHx/GAIN_CHx 寄存器

OS_CHx 与 GAIN_CHx 用来存储相对应通道(x=0 or 1)的 Offset 与 Gain 校准值

Table 5-2 OS_CHx 定义

位置	名称	类型	描述	Default
31:0	OS_CHx	R/W	ADC 通道 x(x=0 or 1)的 Offset 校准值。此校准值可由主控 MCU 端填入，或是在进行 Offset 自校准/系统校准时由芯片自动更新。此校准值为 32bits 有符号补码，高 2 位为符号位。在进行标准模式转换完成后，会先减掉此 Offset 值后再进行增益校准。下面列出各种数值代表意义。 0x00000000: 偏差 0 0x20000000: 正半量程 (+0.50) 0x3FFFFFFF: 正满量程 (+1.00) 0xE0000000: 负半量程 (-0.50) 0xC0000000: 负满量程 (-1.00)	0x00000000

Table 5-3 GAIN_CHx 定义

位置	名称	类型	描述	Default
31:0	GAIN_CHx	R/W	ADC 通道 x(x=0 or 1)的 Gain 校准值。此校准值可由主控 MCU 端填入，或是在进行 Gain 系统校准时由芯片自动更新。此校正值为 32 bit 无符号数。在进行标准模式转换后，会先减掉 Offset 校准值后再乘上此增益校准值。下面列出各种数值代表意义 0x02000000: Gain = 1.00 0x03000000: Gain = 1.50 0x01000000: Gain = 0.5 0x04000000: Gain = 2.00	0x02000000

实际进行 ADC 转换时，一般情况下，即 SYS_CONF0 寄存器的 OGS=0，则 CONV_CONFx 寄存器里的 CHS 选 0 信号信道时，ADC 选取 OS_CH0/GAIN_CH0 里的校准值进行计算。

CONV_CONFx 寄存器里的 CHS 选 1 信号信道时，ADC 选取 OS_CH1/GAIN_CH1 里的校准值进行计算。如表 5-4 前两组所示。

如 SYS_CONF0 寄存器的 OGS=1 时，则 ADC 转换时实际加载的 Offset 和 Gain 校准参数与 CONV_CONFx 寄存器里的 OGSEL 设置有关，而与 CHS 无关。

这样可对转换的同一个信号信道，在不同的转换时刻选用两套不同的 Offset 和 Gain 校准参数。例如在 SYS_CONF0 寄存器的 OGS=1 时，CONV_CONF00 和 CONV_CONF01 的 CHS 都写 '0'，但 OGSEL 分别写 '0' 和 '1'，那么在通过转换命令帧选用 CONV_CONF00 和 CONV_CONF01 进行转换时，所采样转换的信号信道都是信道 0，但采用的 Offset 和 Gain 校准参数则分别为 OS_CH0/GAIN_CH0 和 OS_CH1/GAIN_CH1，如表 5-4 后两组所示。

用户也可以在 MCU 端保存 Offset 和 Gain 校准参数，那么上述 OS_CHx/GAIN_CHx 寄存器都采用默认值即可。

Table 5-4 校准值选择表

SYS_CONF0 寄存器中 OGS 设置	CONV_CONFx 寄存器中的 CHS	CONV_CONFx 寄存器中的 OGSEL	使用的校准值
0	0	X	OS_CH0/GAIN_CH0
0	1	X	OS_CH1/GAIN_CH1
1	X	0	OS_CH0/GAIN_CH0
1	X	1	OS_CH1/GAIN_CH1

5.3.CONV_CONFx 寄存器

CONV_CONFx (x=0 or 1)是用来存储转换设置的寄存器，芯片内共有四组转换设置可供调用，其中 CONV_CONF0 包含 CONV_CONF00 与 CONV_CONF01 两组转换设置，CONV_CONF1 包含 CONV_CONF10 与 CONV_CONF11 两组设置。具体使用哪组转换设置是由转换命令帧中的 CONV_CONF 来决定。在每次开始转换之前，主控 MCU 端需要先将相对应的 CONV_CONFx 寄存器设置好。

表 5-5 列出 CONV_CONFx 的具体定义。

Table 5-5 CONV_CONFx 定义

寄存器	位置	名称	类型	描述	Default
CONV_CONF0	<31:16>	CONV_CONF01	R/W	转换设置 1，具体配置可参考 Table 5-6	0x0000
CONV_CONF0	<15:0>	CONV_CONF00	R/W	转换设置 0，具体配置可参考 Table 5-6	0x0000
CONV_CONF1	<31:16>	CONV_CONF11	R/W	转换设置 3，具体配置可参考 Table 5-6	0x0000
CONV_CONF1	<15:0>	CONV_CONF10	R/W	转换设置 2，具体配置可参考 Table 5-6	0x0000

Table 5-6 CONV_CONFxx 定义

D15	D14	D13	D12	D11	D10	D9	D8
	TMPEN	OGSEL	CHS	OD	DLY	DLY	
D7	D6	D5	D4	D3	D2	D1	D0
DR	DR	DR	DR		GA	GA	GA

位置	名称	类型	描述	Default
15			保留	0x0
14	TMPEN	R/W	温度传感器使能 0: 该次转换的信号信道由 CONV_CONFxx 里的 CHS 决定 1: 该次转换为温度传感器转换 若该次转换为温度传感器转换，则芯片会自动将 SYS_CONF1 中的 TMPPDN 置 1，结束转换后，会自动将 TMPPDN 置 0。	0x0
13	OGSEL	R/W	校准值寄存器选择，此寄存器只有在 SYS_CONF0 的 OGS=1 时才会有作用，具体可以参考 Table 5-4 0: 选择 OS_CH0/GAIN_CH0 1: 选择 OS_CH1/GAIN_CH1	0x0
12	CHS	R/W	待转换的 ADC 通道选择 0: 选择信道 0，对应 VIP0/VIN0 信号引脚 1: 选择信道 1，对应 VIP1/VIN1 信号引脚	0x0
11	OD	R/W	输入端开路检测选择 0: 关闭开路检测功能 1: 开启开路检测功能，芯片内部会输出 0.5uA 电流来检测外部是否开路	0x0

位置	名称	类型	描述	Default
10:9	DLY	R/W	滤波器延时时间选择·用来控制打开 ADC 后到开始进行滤波器转换的时间延迟·1 clock 等于 1 个 ADC 工作时钟周期·默认是 819200Hz。 0x0: 256 ADC clocks 0x1: 512 ADC clocks 0x2: 16 ADC clocks 0x3: 32 ADC clocks	0x0
8			保留	0x0
7:4	DR	R/W	ADC 数据输出码率选择 当 SYS_CONF1 的 FR_SEL = 0 (50 Hz 模式), SYS_CONF0 的 HBF_EN = 0 (不开半带滤波器) 时, 输出频率如下: 0x0: 51200 Hz 0x1: 25600 Hz 0x2: 12800 Hz 0x3: 6400 Hz 0x4: 3200 Hz 0x5: 1600 Hz 0x6: 800 Hz 0x7: 400 Hz 0x8: 200 Hz 0x9: 100 Hz 0xA: 50 Hz 0xB: 25 Hz 0xC: 12.5 Hz 0xD~0xF: 6.25 Hz 当 FR_SEL=1(60 Hz 模式)·输出频率为上述数值乘 1.2 HBF_EN = 1 (开启半带滤波器) 时·上述频率再除以 2	0x0
3			保留	0x0
2:0	GA	R/W	ADC 模拟增益选择 0x0: x64 0x1: x128 0x2: x16 0x3: x32 0x4: x4 0x5: x8 0x6: x1 0x7: x2	0x0

5.4.SYS_CONFx 寄存器

SYS_CONFx (x=0~2)为系统相关的配置寄存器，主控 MCU 端需在上电后先进行系统配置。

5.4.1. SYS_CONF0

Table 5-7 SYS_CONF0 定义

D31	D30	D29	D28	D27	D26	D25	D24
RS_SYS					OGS	HBF_EN	
D23	D22	D21	D20	D19	D18	D17	D16
	CKS_EN						CSHIGH_MODE
D15	D14	D13	D12	D11	D10	D9	D8
PHA	PHA	PHA	PHA	PHA	PHA	PHA	PHA
D7	D6	D5	D4	D3	D2	D1	D0
		OT_S		PW_LV	ERR_CKS	ERR_C	RS_V

位置	名称	类型	描述	Default
31	RS_SYS	R/W	系统全局复位(不包含 SPI 接口部分)，此位被写 1 之后，会进行系统全局复位，此位会在 10 ms 后自动清 0。 芯片有内置上电复位电路，如系统应用允许，可在上电后先写该 BIT 复位芯片，然后进行其他配置。 写入 0: 无效果 写入 1: 开始全局复位 读取 0: 全局复位已完成 读取 1: 全局复位进行中	0x0
30:27			保留	0x0
26	OGS	R/W	校准寄存器选择，具体可参考 Table 5-4 0: 由 CONV_CONFx 寄存器中的 CHS 决定 1: 由 CONV_CONFx 寄存器中的 OGSEL 决定	0x0
25	HBF_EN	R/W	Halfband 滤波器开关选择 0: 关闭 Halfband 滤波器 1: 开启 Halfband 滤波器，此时输出码率会减半	0x0
24:23			保留	0x0
22	CKS_EN	R/W	SPI 界面和校验使能 0: 关闭 SPI 界面和校验 1: 开启 SPI 界面和校验	0x0
21:17			保留	0x0
16	CSHIGH_MODE	R/W	SPI 片选模式选择 0: 关闭 SPI 片选信号可置 1 模式，当芯片处于连续转换模式时，只要 SPI 片选信号置 1，即会退出连续转换模式 1: 开启 SPI 片选信号可置 1 模式，当连续转换模式时，当 SPI 片选置 1，不会退出连续转换模式。当 SPI 片选再度置 0 时，会继续之前的连续转换模式，直到收到停止连续转换模式指令(详见转换命令帧章节)。	0x0
15:8	PHA	R/W	ADC 码流相位延时选择 0x00: 不延时 0x01: 延后 1 个 ADC 时钟 0x02: 延后 2 个 ADC 时钟 ... 0xFF: 延后 255 个 ADC 时钟	0x0
7:6			保留	0x0

位置	名称	类型	描述	Default
5	OT_S	R	Offset 自校准标志位，此位与 SYS_CONF1 的 SHI 位同步变化 0: Offset 自校准未进行 1: Offset 自校准进行中	0x0
4			保留	0x0
3	PW_LV	R	芯片电源欠压检测 0: 芯片电源在 3V 以上 1: 芯片电源已掉到 3V 以下	0x0
2	ERR_CKS	R	SPI 同位或和校验错误，此寄存器会在主控 MCU 读取此寄存器后自动清 0 0: 未发生错误 1: 表示最近一次接收到的命令帧同位错误，或是写入的 32bits 数据和校验错误。	0x0
1	ERR_C	R	ADC 转换错误，此寄存器会在主控 MCU 读取此寄存器后自动清 0 0: 未发生错误 1: ADC 转换错误 有两种错误可能 a. 未转换完成(SDO 未变为 0)前就被打断 b. ADC 转换结果发生溢出(overflow)	0x0
0	RS_V	R/W	复位有效标志 0: 上次复位失败，主控 MCU 需要重新进行全局复位 1: 上次复位成功	0x0

5.4.2. SYS_CONF1

Table 5-8 SYS_CONF1 定义

D31	D30	D29	D28	D27	D26	D25	D24
RCHPDN	POWD	FR_SEL	SHI	VRS		CHS	OD
D23	D22	D21	D20	D19	D18	D17	D16
	GA<2>	GA<1>	GA<0>			TMPPDN	ADCPDN
D15	D14	D13	D12	D11	D10	D9	D8
GAIN_MODE				ADITB<1>	ADITB<0>	ADITA<1>	ADITA<0>
D7	D6	D5	D4	D3	D2	D1	D0
IDT<1>	IDT<0>	CKMODE<1>	CKMODE<0>				

位置	名称	类型	描述	Default
31	RCHPDN	R/W	内部高频 RC 时钟开关，此寄存器受晶体时钟监测结果控制，如果监测到连续 200ms 没有晶体时钟，则会自动将此寄存器置 1，打开 RCH 时钟并将系统时钟切换至 RCH。 读取 0: RCH 模块已被关闭 读取 1: RCH 模块已被打开 主控 MCU 端也可以强制打开 RCH 模块 写入 0: 无作用 写入 1: 强制打开 RCH 模块，但是并不会将系统时钟切换到 RCH	0x0
30	POWD	R/W	低功耗模式选择 0: 正常模式 1: 低功耗模式，此模式下芯片功耗小于 1uA	0x0
29	FR_SEL	R/W	频率模式选择 0: 50Hz 模式，对应 ADC 时钟 819.2kHz (系统频率除以 6) 1: 60Hz 模式，对应 ADC 时钟 983.04kHz (系统频率除以 5)	0x0

位置	名称	类型	描述	Default
28	SHI	R/W	输入信号短路·该 BIT 与 SYS_CONF0 里的 OT_S 同步变化 读取 0: 内部短路功能已关闭 读取 1: 内部短路已使能 写入 0: 无作用 写入 1: 强制使能内部短路	0x0
27	VRS	R/W	ADC 基准源选择 0:使用外部基准源·VREF+和 VREF-为基准源输入 1:使用内部 BGP 电路作为基准源·基准电压约 1.2V·典型温度系数 10ppm/°C·从 VREF+和 VREF-引脚输出。外部 VREF+和 VREF-之间需加 1 个 1uF 去耦电容。	0x0
26			保留	0x0
25	CHS	R	ADC 通道选择·此寄存器在开始 ADC 转换时·会自动更新为 CONV_CONFx 寄存器内的 CHS 设定值 0: 选择 ADC 通道 0 1: 选择 ADC 通道 1	0x0
24	OD	R	ADC 开路检测选择·此寄存器在开始 ADC 转换时·会自动更新为 CONV_CONFx 寄存器内的 OD 设定值 0: 关闭开路检测功能 1: 开启开路检测功能	0x0
23			保留	
22:20	GA	R	ADC 增益选择·此寄存器在开始 ADC 转换时·会自动更新为 CONV_CONFx 寄存器内的 GA 设定值 0x0: x64 0x1: x128 0x2: x16 0x3: x32 0x4: x4 0x5: x8 0x6: x1 0x7: x2	0x0
19:18			保留	0x0
17	TMPPDN	R/W	温度传感器使能·此寄存器在开始 ADC 转换时·会自动更新为 CONV_CONFx 寄存器内的 TMPEN 设定值 0: 温度传感器已关闭 1: 温度传感器已使能	0x0
16	ADCPDN	R/W	ADC 模拟模块使能·上电后主控 MCU 需设置该 BIT 为' 1' 以打开 ADC 读取 0: ADC 已关闭 读取 1: ADC 已使能 写入 0: 无作用 写入 1: 使能 ADC	0x0
15	GAIN_MODE		当 ADC 增益选择为小于等于 16 倍时·需将此 BIT 置' 1'·大于 16 倍时为默认的' 0'	0x0
14:12			保留	0x0
11:10	ADITB<1:0>	R/W	ADC 电路 B 部分的电流消耗设置·在需要降低功耗的应用场合可配置此寄存器为 01。 00: 默认功耗 01: -50% 10: 禁止 11: 禁止	0x0

位置	名称	类型	描述	Default
9:8	ADITA<1:0>	R/W	ADC 电路 A 部分的电流消耗设置。正常情况下需设置为 10。在需要降低功耗的应用场合，可在降低 ADC 工作频率的基础上（见 SYS_CONF2），配置此寄存器为 01。 00: 默认功耗 01: -50% 10: +50% 11: 禁止	0x0
7:6	IDT<1:0>	R/W	仪表放大器的电流消耗设置。在需要降低功耗的应用场合可配置此寄存器为 01 或 10。 00: 默认功耗 01: -33% 10: -50% 11: -60%	0x0
5	CKMODE<1>	R/W	在 51.2kHz 的码率设置时，不管发起的是单次转换还是连续转换模式，该位必须配置为 1。如果发起的是单次转换模式，800Hz 以上的码率设置时，该位也必须配置为 1。	0x0
4	CKMODE<0>	R/W	需配置为 1	0x0
3:0			保留	0x0

5.4.3. SYS_CONF2

Table 5-9 SYS_CONF2 定义

D31	D30	D29	D28	D27	D26	D25	D24
ADCKDIV2		REFTRIM_L					
D23	D22	D21	D20	D19	D18	D17	D16
	REFTRIM	REFTRIM	REFTRIM			DMODE <1>	DMODE <0>
D15	D14	D13	D12	D11	D10	D9	D8
			BGP_MODE	RCTRIM<3>	RCTRIM<2>	RCTRIM<1>	RCTRIM<0>
D7	D6	D5	D4	D3	D2	D1	D0
	VCMTRIM						

位置	名称	类型	描述	Default
31	ADCKDIV2	R/W	ADC 工作频率选择。正常情况下采用默认值即可。该 BIT 设 1 后，CONV_CONFx 寄存器里 DR 所对应的 ADC 数据率也会相应除以 2。在 ADC 精度足够，但功耗更为敏感的应用领域，可将该 BIT 设为 1 后，再将 ADITA<1:0> 设置为 01，以及 IDT<1:0> 设置为 10，以降低芯片功耗。此时 ADC 的 ENOB 将下降约 1BIT。 0: 819.2kHz (50Hz 模式下，60Hz 模式下则乘 1.2) 1: 409.6kHz	0x0
30			保留	0x0
29	REFTRIM_L	R/W	内部基准电压源(BGP)的温度系数调节 0: 默认值 1: +45ppm/°C	0x0
28:23			保留	0x0
22:20	REFTRIM<2:0>	R/W	内部基准电压源(BGP)的温度系数调节 000:默认值; 001:+10ppm/°C; 010:+20ppm/°C; 011:+30ppm/°C; 100:+30ppm/°C; 101:-30ppm/°C; (建议设定值) 110:-20ppm/°C; 111:-10ppm/°C	0x0

位置	名称	类型	描述	Default
19:18			保留	0x0
17:16	DMODE	R/W	DMODE 设置。在资料率 DR 设置为 400Hz 及以下时，建议配置为' 10'；DR 在 400Hz~6.4kHz 时，配置为' 00'；DR 在 12.8kHz 及以上时，配置为' 11'。 00:12.8k; 01:6.4k;10:800; 11:无	
15:13			保留	0x0
12	BGP_MODE	R/W	内部基准电压源(BGP)的模式。在使用内部 BGP 作为 ADC 基准源时，建议配置为 1 0:默认模式 1:低 offset 模式	0x0
11:8	RCTRIM<3:0>	R/W	高频 RC 时钟的频率调节。 芯片内部 RCH 时钟在-40~85 度范围内随温度的频率变化小于 1%，但芯片之间的 RCH 频率存在固有偏差。如系统应用在无晶体的场合，且希望 RCH 频率精度较高时，需要 MCU 做额外校正，校正正值配置到此寄存器。有晶体的应用场合则无需配置此寄存器。 0000:默认值; 0001:-2.5%; 0010: -5%; 0011: -7.5%; 0100: -10%; 0101: -12.5%; 0110: -15%; 0111: -17.5% 1000: +20%; 1001: +17.5%; 1010: +15%; 1011: +12.5%; 1100: +10%; 1101: +7.5%; 1110: +5%; 1111: +2.5%	0x0
7			保留	0x0
6	VCMTRIM	R/W	输入信号共模电压调节 0: 默认值 1: 调节输入信号共模电压，在输入信号共模低于 0.5V 时配置为' 1'	0x0
5:0			保留	0x0

5.5.D_TARG 寄存器

D_TARG 寄存器为系统 GAIN 校准时需要用到的寄存器，详见 GAIN 系统校准章节。

Table 5-10 D_TARG 定义

位置	名称	类型	描述	Default
31:0	D_TARG	R/W	系统 Gain 校准之 ADC 期望值 0x20000000: 正半量程 (+0.50) 0x3FFFFFFF: 正满量程 (+1.00) 0xE0000000: 负半量程 (-0.50) 0xC0000000: 负满量程 (-1.00)	0x3FFFFFFF

5.6.CONV_DATA 寄存器

CONV_DATA 寄存器为 ADC 转换完成后数据存储的寄存器，通过转换命令帧后获得的 ADC 转换数据除了存储在 CONV_DATA 之外，同时也会及时通过 SPI 送给主控 MCU。一般情况下，发送转换命令帧后需及时读取 ADC 数据，详见转换命令帧章节

Table 5-11 CONV_DATA 定义

位置	名称	类型	描述	Default
31:1	DATA	R	ADC 转换结果 bit[31:1] · bit[0]为当次数据所的信号信道。bit[31:1]为 31 位有符号补码，高 2 位为符号位，将 bit[0]补 0 后，bit [31:0]的结果: 0x20000000: 正半量程 (+0.50) 0x3FFFFFFF: 正满量程 (+1.00) 0xE0000000: 负半量程 (-0.50) 0xC0000000: 负满量程 (-1.00)	--
0	CHL	R	表示该次转换的通道选择 0: 该次转换为通道 0 1: 该次转换为通道 1	--

6. ADC 校准

6.1. 校准概述

芯片的校准分为两个部分，offset 校准和 gain 校准。

同时又分为自校准和系统校准两类，不管是自校准还是系统校准，用户都需先校准 offset，后校准 gain。

校准时也沿用 CONV_CONFx 寄存器里的 DR (Data Rate)设置，如果时间允许，客户应尽可能使用更低的数据率来进行 offset 校准和 gain 校准，以便得到更精确的校准值。如果期望校准值再精确（如 24BIT 以上的无噪声精度），可以由主控 MCU 端多发起几次校准，将几次的校准值取平均值，再写入校准寄存器中。

6.2. Offset 自校准

MCU 通过转换命令帧配置芯片进入 offset 自校准模式后（详见 4.3. 转换命令帧章节），芯片将在内部将选中通道的输入端短路，此时 ADC 输入的信号为 0 信号，然后以指定的 CONV_CONFx 寄存器中的 ADC 配置参数（如 ADC 增益、DR 配置等）测量此时的 ADC 值，所测得的 ADC 转换值即为芯片自身所具有的 offset，芯片自动将该 offset 值填入相应通道的 offset 校准寄存器中，同时也从 SDO 上输出数据以便 MCU 读取。

6.3. Offset 系统校准

MCU 通过转换命令帧配置芯片进入 offset 系统校准模式后，芯片将以指定的 CONV_CONFx 寄存器中的 ADC 配置参数（如 ADC 增益、DR 配置等）测量此时的 ADC 值。在此之前，使用者需保证芯片外部所加信号为 0，此时所测得的值即为系统 offset，芯片自动将该 offset 值填入相应通道的 offset 校准寄存器中，同时也从 SDO 上输出数据以便 MCU 读取。

MCU 也可通过转换命令帧配置芯片进入单次或连续转换模式，开启测量。在此之前，使用者需保证芯片外部所加信号为 0，此时所测得的值即为系统 offset。然后 MCU 通过 SPI 读取 ADC 值，如需要更高精度，可多次测量，将这几次的 offset 求平均后，再写入芯片相应通道的 offset 校准寄存器中。

6.4. Gain 系统校准

增益(gain)的系统校准由 MCU 实现，需在校准 offset 之后实施。校准 Gain 时，使用者需保证芯片外部所加信号幅度达到满量程的 10%以上（最好为 20%~50%之间），再由 MCU 发起转换操作，芯片将以指定的 CONV_CONFx 寄存器中的 ADC 配置参数（如 ADC 增益、DR 配置等）测量此时的 ADC 值。

转换完成后，MCU 通过 SPI 读取 ADC 值，如需要更高精度，可多次测量，将这几次的值求平均得到 D_{out} ，再根据当前所加外部信号对应的目标值 D_{targ} （即用户希望当前信号所对应的 ADC 值），计算增益校准系数 $GAIN_CHx = D_{targ}/D_{out}$ ，并将 GAIN_CHx 填入相应通道的 gain 校准寄存器中。

6.5. 正常转换时的数据校准

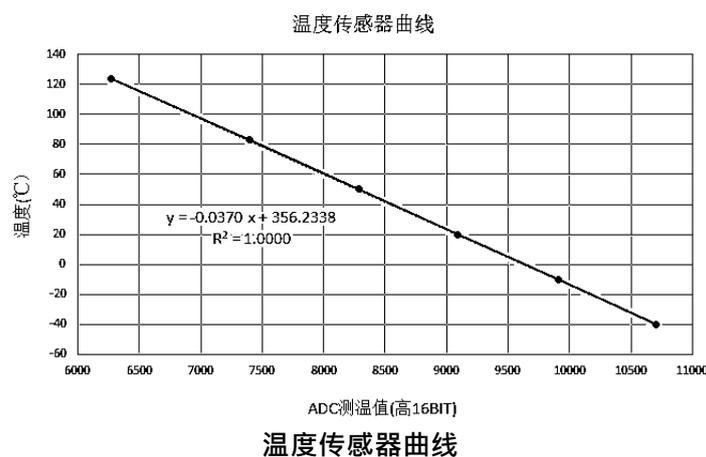
正常转换时，假设 ADC 经滤波器得到的值为 D_{ori} ，根据 CONV_CONFx 寄存器里的配置（如果系统寄存器里的 OGS= '0'，则使用信号信道序号所对应的 Offset/Gain 校准寄存器组的值），该次转换对应选择的 offset 和 gain 校准寄存器值分别为 OS_CHx 和 GAIN_CHx，则芯片将自动计算值 $D_{out} = GAIN_CHx * (D_{ori} - OS_CHx)$ ，并将其填入转换数据寄存器。

7. 温度传感器

芯片内部集成 2°C 精度的温度传感器，通过设置 CONV_CONFx 寄存器里的 TMPEN=1，即可在以该 CONV_CONFx 作为转换配置时，对芯片温度进行测量。

传感器曲线的曲线：

只使用 ADC 的高 16BIT 进行计算，此时的公式如下所示：



公式为 $y = -0.037x + 356.23$ ，其中 x 为高 16BIT ADC 资料，求得的 y 即为当前温度。公式中的截距系数 356.23，对于每颗芯片来说会有一定的差异，需要在应用时做单点校正，即根据已知温度 y_0 和 ADC 值 x_0 ，计算当前芯片的准确截距值。增益系数 -0.037，芯片之间的差异较小，不用额外做校正。

8 ADC 测试图表

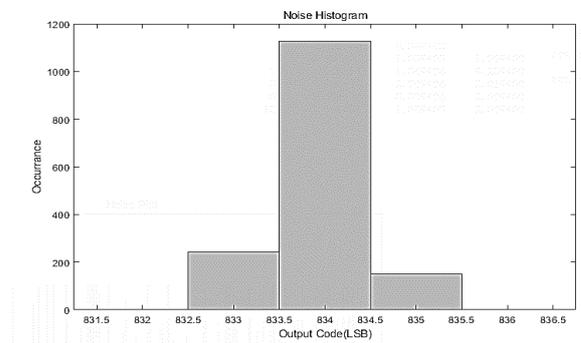
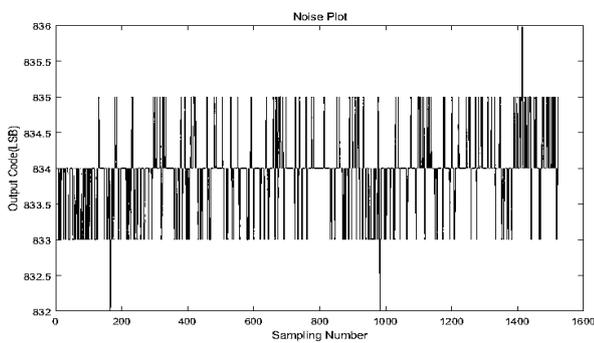
8.1 不同增益设置下的噪声值

测试条件：

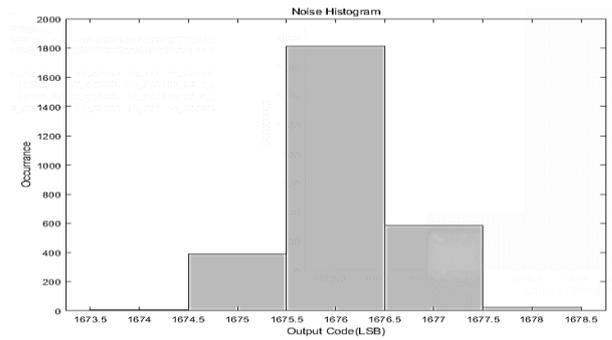
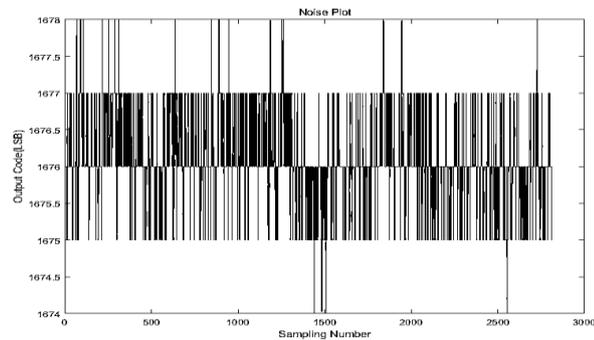
输入 0.5mV 的直流信号， $AV+=DV+=VREF+=5V$ ， $VREF-=GND$ ，采样率 6.25Hz。

为便于对比，图中的数据只取了高 25BIT。由于最高位为一个冗余的符号位，因此高 25BIT，实际等效于常见的 24BIT 数据。

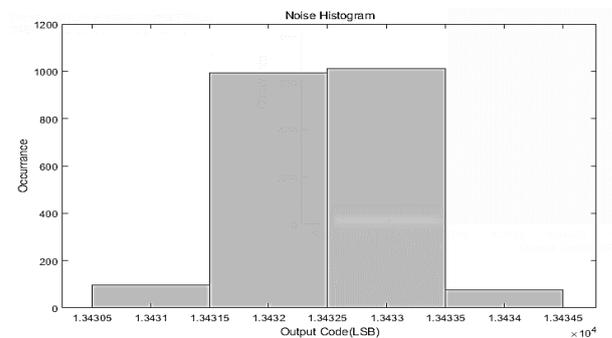
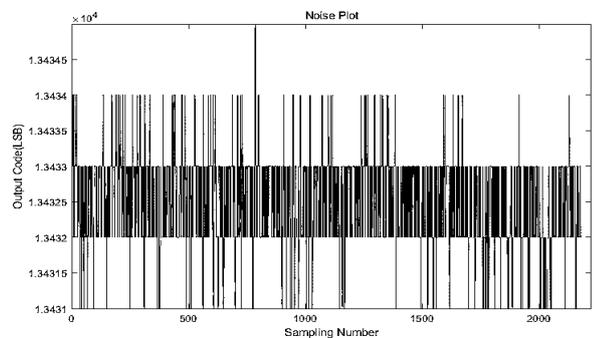
PGA=1



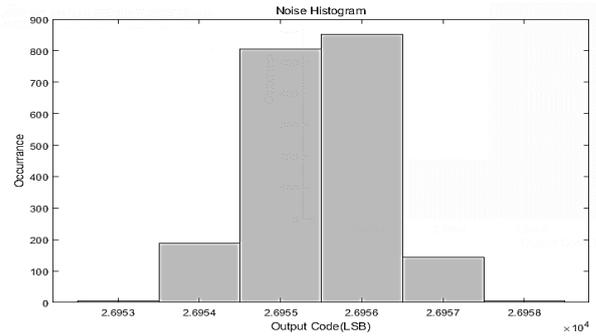
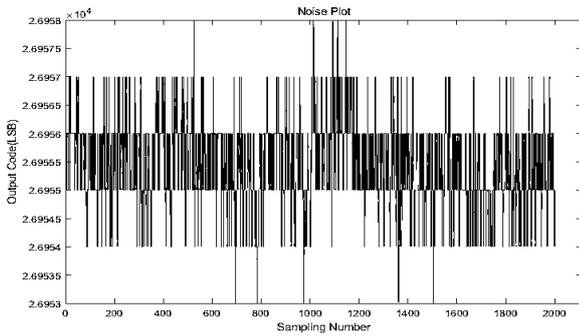
PGA=2



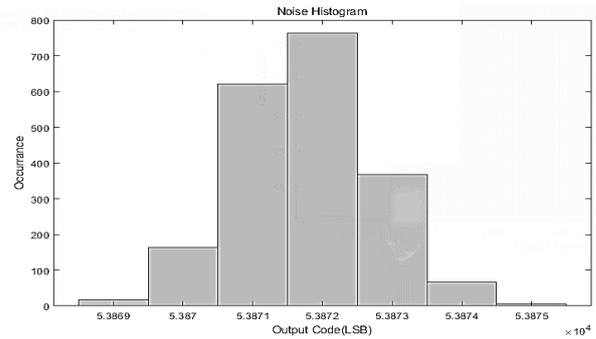
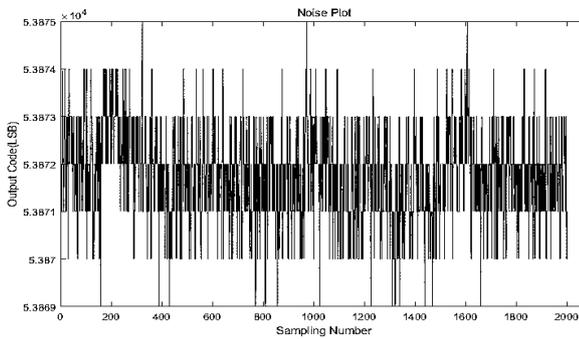
PGA=16



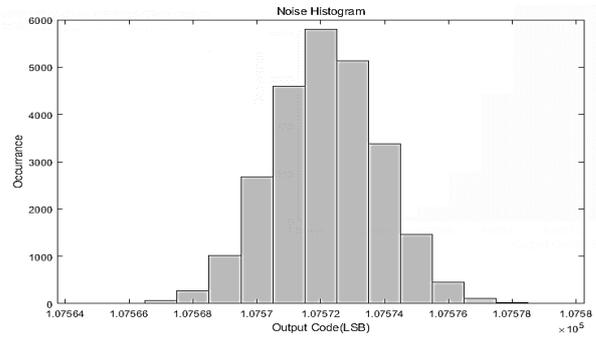
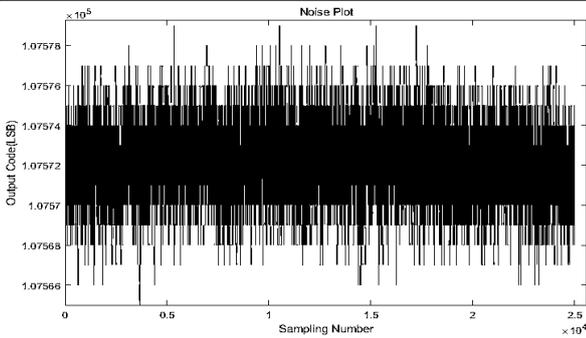
PGA=32



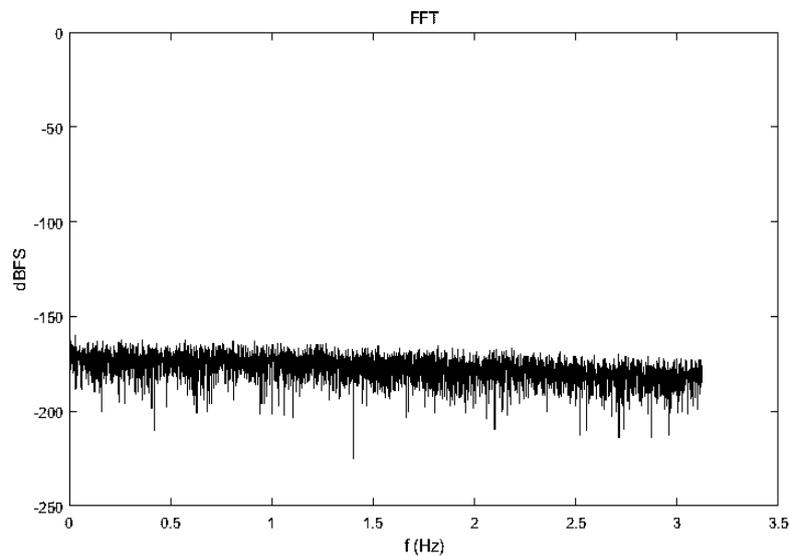
PGA=64



PGA=128



下图为 PGA=128 时的信号频谱图，无谐波、无 Flicker noise。

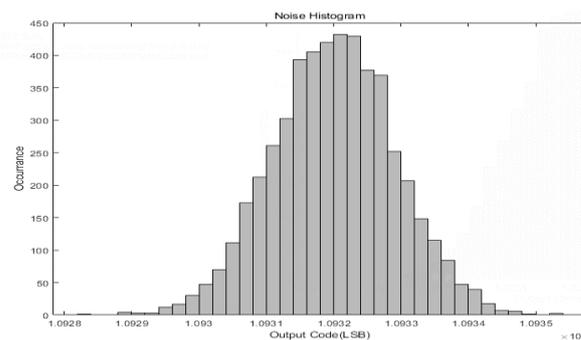
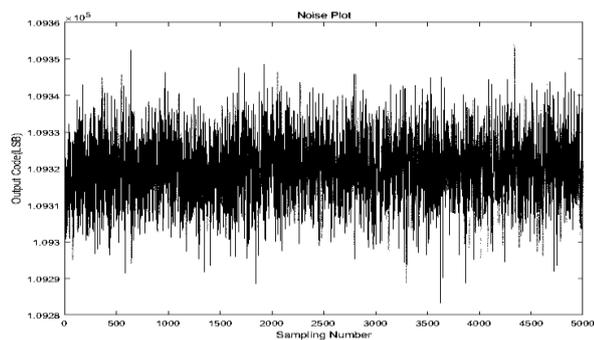


8.2 不同数据码率设置下的噪声值

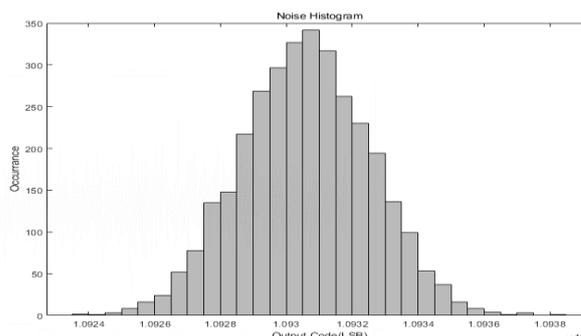
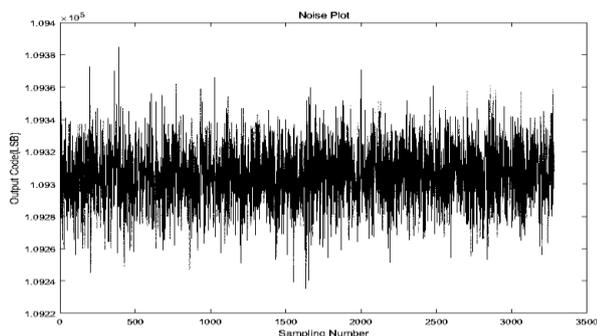
测试条件：

输入 0.5mV 的直流信号， $AV+=DV+=VREF+=5V$ ， $VREF-=GND$ ，增益固定为 128 倍。
为便于对比，图中的数据只取了高 25BIT。由于最高位为一个冗余的符号位，因此高 25BIT，实际等效于常见的 24BIT 数据。

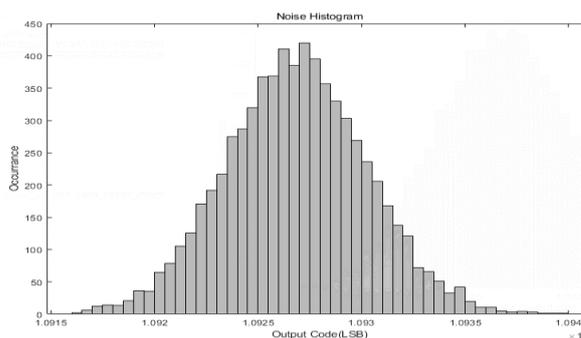
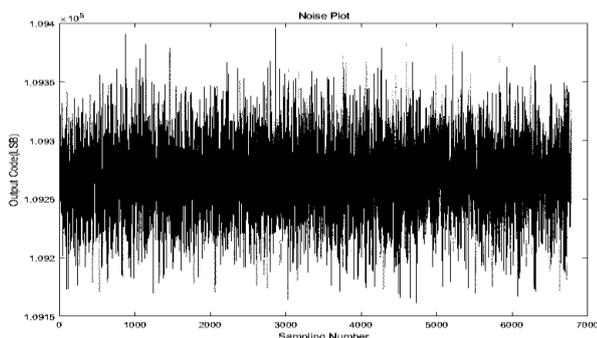
DR=200Hz



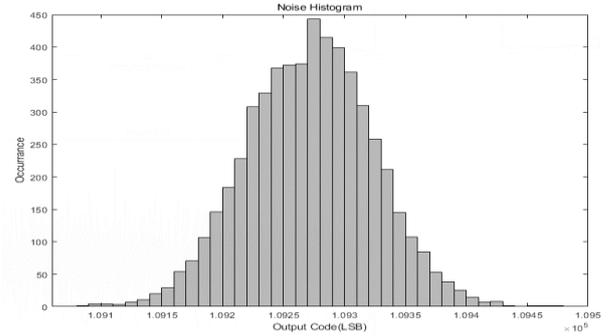
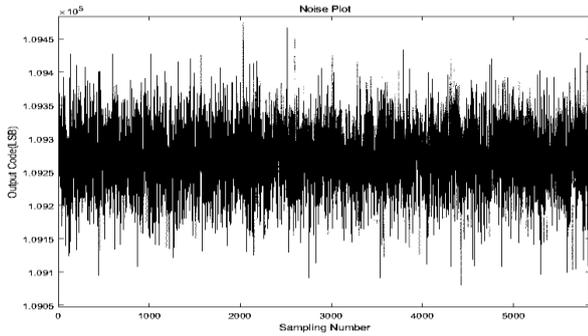
DR=800Hz



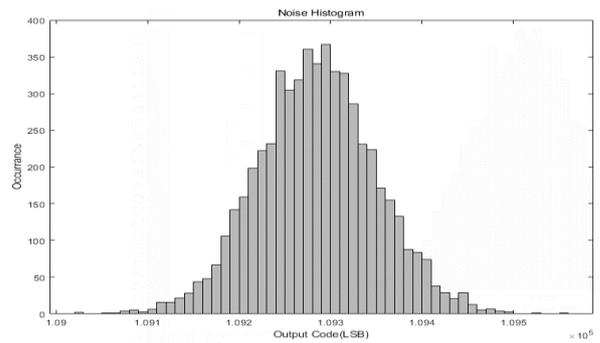
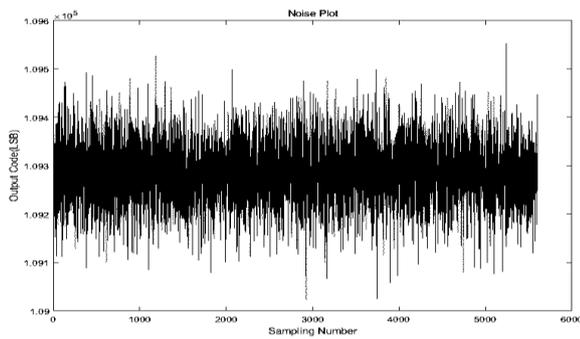
DR=3200Hz



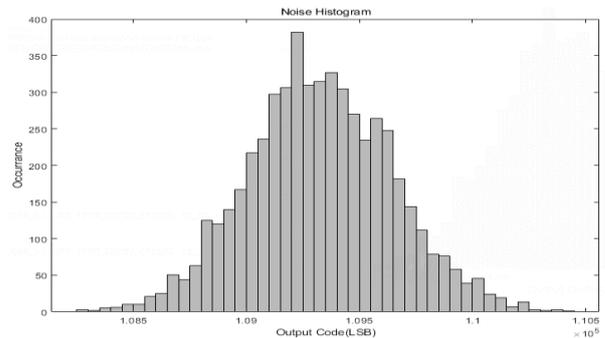
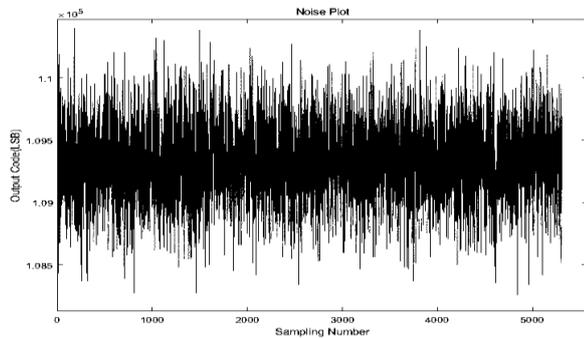
DR=12800Hz



DR=25600Hz



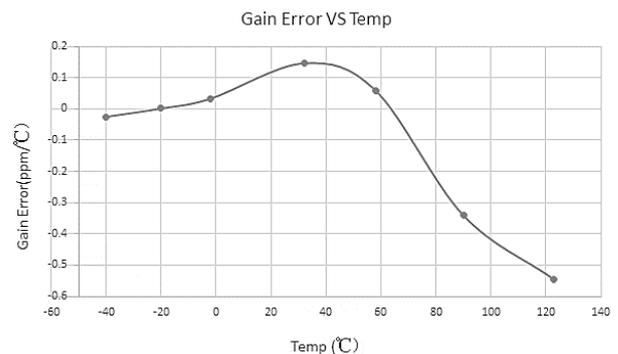
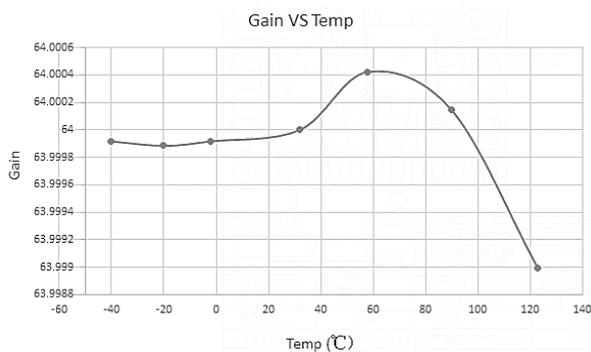
DR=51200Hz



8.3 增益和 Offset 的温漂

增益温漂测试条件：

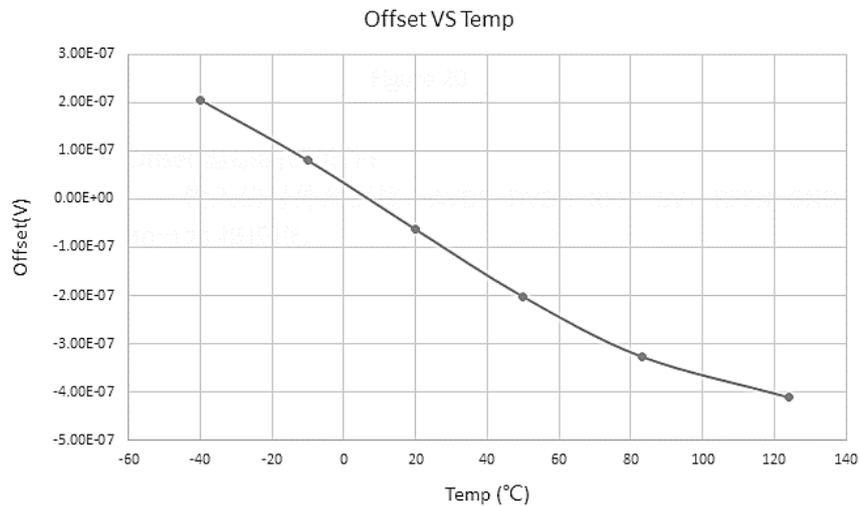
输入一半满量程的直流信号， $AV+=DV+=VREF+=5V$ ， $VREF-=GND$ ，增益设置为 64 倍， $DR=6.25Hz$ 。测试温度范围-40~125 摄氏度。



Offset 温漂测试条件：

输入信号外部短接， $AV+=DV+=VREF+=5V$ ， $VREF-=GND$ ，增益设置为 64 倍， $DR=6.25Hz$ 。

测试温度范围-40~125 摄氏度。



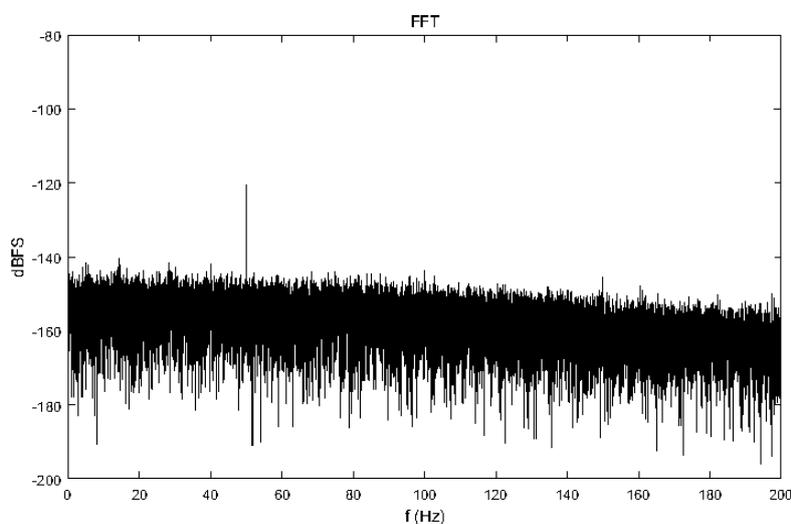
8.4 电源抑制比(PSRR)

测试条件：

电源上迭加 1V 峰峰值、50Hz 的正弦信号， $AV+=DV+=4.5V$ (即电源在 4~5V 之间变化)， $VREF+=1.2V$ (内部基准电压源产生)， $VREF-=GND$ ，PGA 设置为 64 倍，采样率 400Hz。频谱图如下图所示。

电源工频干扰信号在输出数据上为-120dB，即 1 μ V，等效到输入端为 $1\mu V \cdot 1.2V / 64 = 18.8nV$ (其中 1.2V 为基准电压)。1V 峰峰值的输入电源干扰的有效值为， $1/2/1.414 = 0.354V$ 。

则 PSRR 计算为： $20 \cdot \log_{10}(0.354/18.8n) = 145dB$



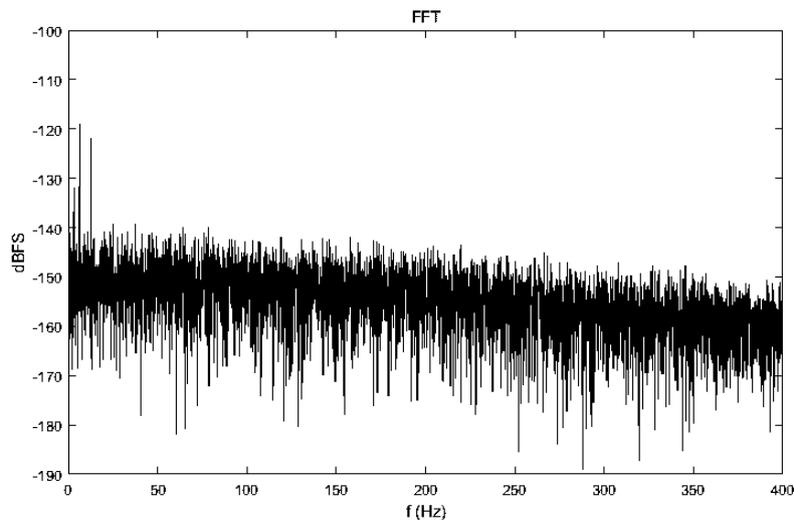
8.5 共模抑制比(CMRR)

测试条件：

在输入信号上迭加 2V 峰峰值、6.25Hz 的正弦共模信号， $AV+=DV+=VREF+=5V$ ， $VREF-=GND$ ，PGA 设置为 128 倍，采样率 800Hz。频谱图如下图所示。

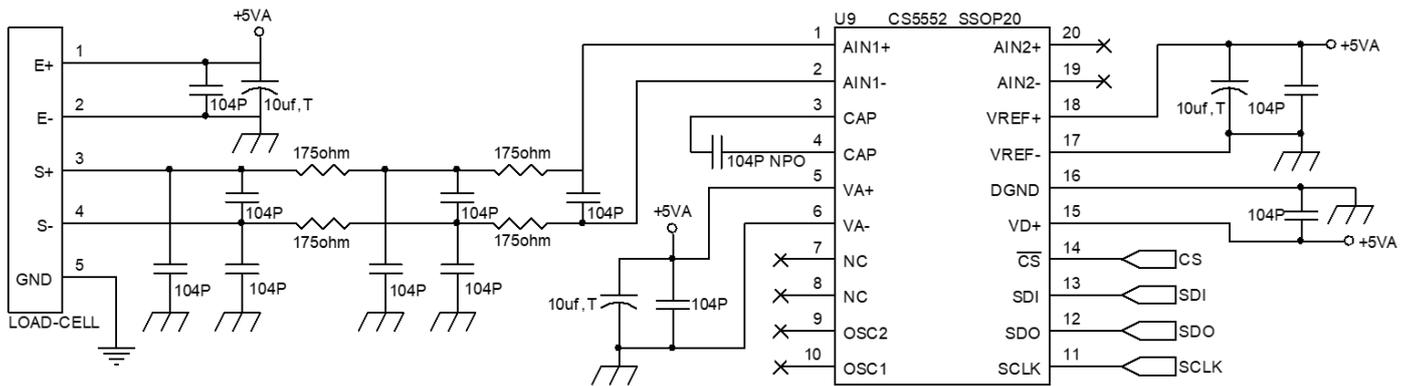
共模干扰信号在输出数据上为 -119dB，即 1.1 μ V，等效到输入端为 $1.1\mu V \cdot 5V / 128 = 43.8nV$ （其中 5V 为基准电压）。2V 峰峰值的输入共模干扰的有效值为， $2/2/1.414 = 0.707V$ 。

则 CMRR 计算为： $20 \cdot \log_{10}(0.707/43.8n) = 144dB$

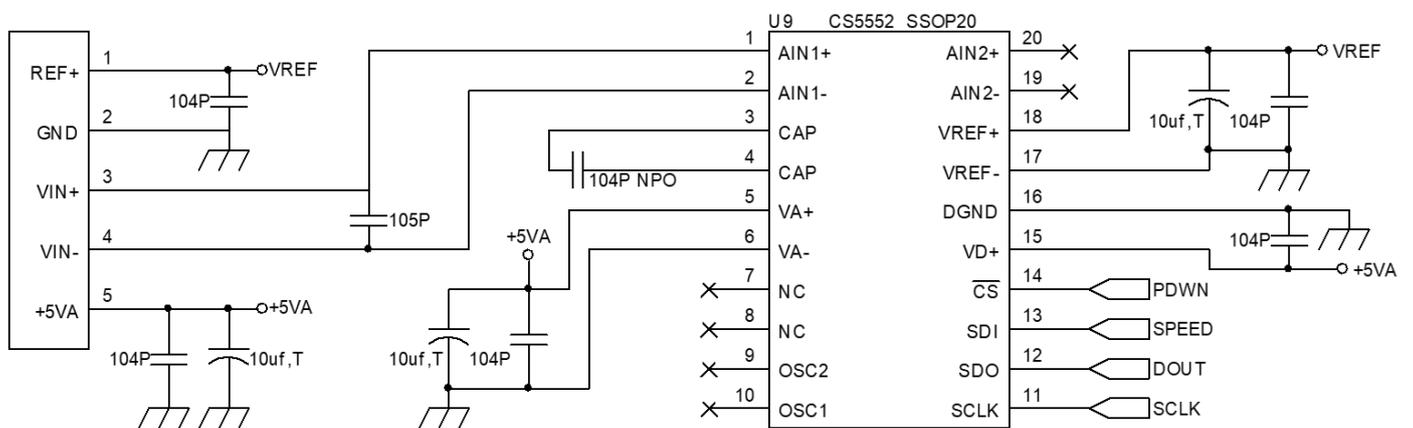


9. 参考应用电路

差分输入应用



单端输入应用



10. PACKAGING 封装

20 PIN SSOP PACKAGE DRAWING

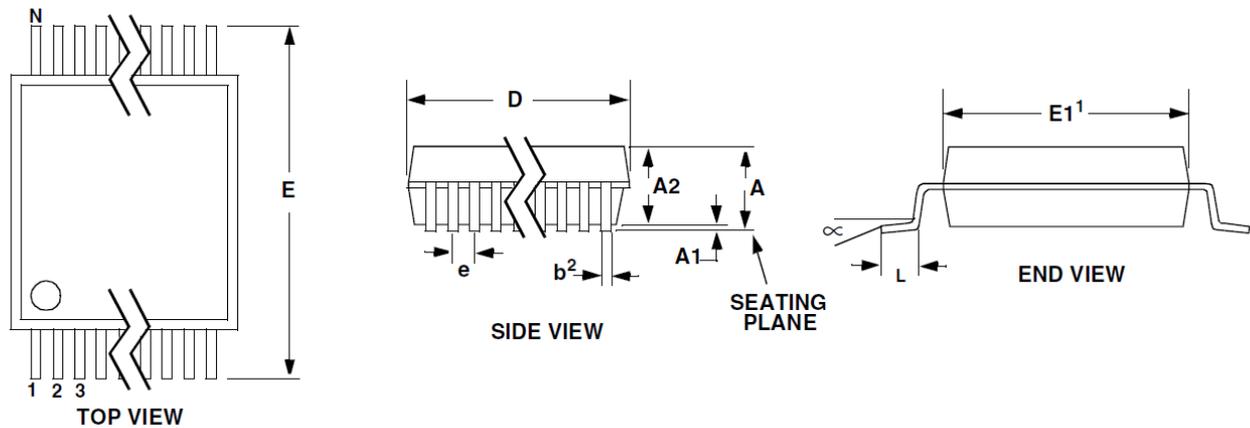


Figure12.

DIM	INCHES		MILLIMETERS		NOTE
	MIN	MAX	MIN	MAX	
A	--	0.084	--	2.13	
A1	0.002	0.010	0.05	0.25	
A2	0.064	0.074	1.62	1.88	
b	0.009	0.015	0.22	0.38	2,3
D	0.272	0.295	6.90	7.50	1
E	0.291	0.323	7.40	8.20	
E1	0.197	0.220	5.00	5.60	1
e	0.024	0.027	0.61	0.69	
L	0.025	0.040	0.63	1.03	
∞	0°	8°	0°	8°	

- Notes:
1. "D" and "E1" are reference datums and do not include mold flash or protrusions, but do include mold mismatch and are measured at the parting line, mold flash or protrusions shall not exceed 0.20 mm per side.
 2. Dimension "b" does not include dambar protrusion/intrusion. Allowable dambar protrusion shall be 0.13 mm total in excess of "b" dimension at maximum material condition. Dambar intrusion shall not reduce dimension "b" by more than 0.07 mm at least material condition.
 3. These dimensions apply to the flat section of the lead between 0.10 and 0.25 mm from lead tips.

11.应用开发笔记

硬件注意参考事项

- 如使用芯片内部基准电压源，REFP 和 REFN 之间接一个 1uF 电容，REFN 接 AVSS。
- 如果 ADC 应用于采样率 200Hz 以上，则 CAP 脚浮空，或者选用容值更小的电容，例如 1nF
- 如输入信号可能受高频电磁场干扰，则需在信号输入端加 RC 滤波电路，RC 常数的设置，与所需的信号带宽相当即可，先确定电阻值，再根据带宽确定电容值。其中电阻 R 的值应优先选用 100 欧，因为阻值选取过大的话，电阻的热噪声就将比整个 ADC 芯片的噪声要大了。假设选用 1k 的电阻，则两端 2k，其热噪声为 $5.8nV/\sqrt{\text{Hz}}$ ，高于芯片的 $3.1nV/\sqrt{\text{Hz}}$ 的噪声。

软件配置流程

11.1 配置寄存器

芯片上电后，上位机通过 SPI 接口发送写命令帧，往 OS_CHx/ GAIN_CHx(x=0/1)寄存器、CONV_CONFx(x=0/1)寄存器、SYS_CONFx(x=0/1/2)寄存器写需要的设置值。

11.1.1 OS_CHx/ GAIN_CHx(x=0/1)寄存器

一般来说，ADC 的偏置 OS 和增益 GAIN 要从系统层面进行校正。最初的时候，往寄存器里写默认值，默认值分别为 0x00000000 (OS) 和 0x02000000 (GAIN)。在完成配置工作后，再进行校正操作。

11.1.2 CONV_CONFx(x=0/1)寄存器

配置 ADC 转换工作时的设置。一般而言，一个信号信道对应一条转换设置。

GA : 根据最大信号 Vmax 的大小、REF 值进行选择。要求是 $V_{\text{max}} * \text{GAIN} < \text{REFP} - \text{REFN}$ (内部 REF 的话， $\text{REFP} - \text{REFN} = 1.2V$)

DR : 根据需求进行选择

DLY : 一般默认 0 即可

CHS : 根据硬件所用信号信道进行选择

OGSEL : 只有在使用同一个信号信道，但是需要使用两套 OS/GAIN 校正寄存器的时候才需要配置。详见手册说明

TMPEN : 如果设置为 1，则该转换指令用于测量温度传感器。

10.1.3 SYS_CONFx(x=0/1)寄存器

系统寄存器里需要配置的如下所示：

SYS_CONF0

HBF_EN : 开启 Halfband 滤波器后，输出码率会减半，但信号带宽内的通带平坦度更好。例如希望设置为 3200Hz 的数据码率，可以直接通过 DR 设置为 3200Hz，也可以设置 DR 为 6400Hz，然后开启 HBF_EN。对于单次转换的应用，或者采样信号为直流的应用，不需要开启 HBF_EN。

CSHIGH_MODE : 参看资料手册说明。在一路 SPI 同时读写 2 颗以上芯片的时候，必须将该 BIT 置' 1'。此外，一路 SPI 对应一颗芯片的时候，如果现场电磁环境较差，导致 ADC 连续转换模式的数据连续读取经常出错，也可以将该 BIT 置' 1'

PHA : 相位延时选择，用于校正信号相位差。

CKS_EN : SPI 接口和校验使能，开启可让 SPI 通讯更可靠，默认为不开启。如选择开启和校验，可在配置所有寄存器之前，先配置这一 BIT，然后再配置其他寄存器。**建议开启和校验功能**

RS_SYS : 芯片内部有上电复位功能，可以实现可靠复位。但假如芯片功能不正常，或者 SPI 通讯不上，则应先复位 SPI 模块 (参考数据手册)，然后将 RS_SYS 位写' 1'，对芯片进行系统复位

OT_S/ PW_LV/ ERR_CKS/ ERR_C/ RS_V 这几个 BIT 为状态 (标志) 寄存器，不能进行写入操作。

SYS_CONF0 的其他寄存器位根据应用需求进行设置

SYS_CONF1

TMPPDN/GA/OD/CHS 寄存器无需配置，芯片会根据 CONV_CONFx(x=0/1)寄存器里的配置自动打开相应的控制 BIT

ADCPDN : 需配置为' 1' 打开 ADC 模块

VRS : ADC 基准电压选择。如果选择内部基准电压 REF，则 REFP=1.2V，此时推荐设置 BGP_MODE=' 1'

SHI: 信号内部短路，可用于 ADC offset 校准。

FR_SEL : 50/60Hz 频率选择，根据所在国家电网频率选择

POWD : 选择正常工作模式还是睡眠模式，睡眠模式下会将大部分模块关闭。

GAIN_MODE : 当 ADC 增益选择为小于等于 16 倍时，需将此 BIT 置' 1'，大于 16 倍时为默认的' 0'

CKMODE<0> : 需配置为' 1'

CKMODE<1> : 800Hz 以上的码率设置时，该位也必须配置为 1

ADITA<1:0> : 在正常模式下需设置为' 10'。

IDT<1:0> : 对功耗没有要求的场合无需配置,保持默认的00即可。在ADC精度足够,但功耗更为敏感的应用领域,可将ADCKDIV2位设为1后,再将ADITA<1:0>设置为01,以及IDT<1:0>设置为10,以降低芯片功耗。此时ADC的ENOB将下降约1BIT。

SYS_CONF2

ADCKDIV2 : 根据功耗要求决定是否需要配置为1,详见手册说明。

REFTRIM_L/REFTRIM<2:0> : 用于修正内部REF模块的温度系数。使用外部REFF/REFN引脚接入的应用里,无需配置。使用内部REF的应用里,根据温度系数的要求进行配置。推荐配置为REFTRIM<2:0>='101'。

DMODE<1:0> : DMODE设置,在资料率DR设置为400Hz及以下时,建议配置为'10';DR在800Hz~6.4kHz时,配置为'00';DR在12.8kHz及以上时,配置为'11'。

RCTRIM<3:0> : 内部高频RC时钟频率调节寄存器,在无外部晶体/无外部输入时钟源的应用里,如果希望内部高频RC时钟有更精确的频率,需要测量每颗芯片的实际频率,并填写相应的RCTRIM<3:0>寄存器值。

VCMTRIM : 输入信号共模电压调节,如信号共模低于0.5V,则需将此位设置为'1'。

11.1.4 连续转换模式下的注意事项

如ADC芯片处于连续转换模式下,由于会处于长时间工作,期间SPI接口可能受到强干扰而导致数据读取连续出错。

此时需要有途径获知SPI读到的ADC数值已出错,因此强烈建议将CKS_EN置'1',打开和校验功能,如和校验发现资料连续出错(如连续三次),则复位SPI接口,然后重新发起连续转换命令帧。如只错误一个数据,则仅丢弃错误数据。打开校验功能后的SPI时序图,详见数据手册相应章节。

另一种方法是,将CSHIGH_MODE置'1',在每读完一个资料后,将CSN引脚拉高一次。此时如SPI有异常状态,CSN拉高期间将被清除。CSHIGH_MODE置'1'的另一个用途是MCU可通过一个SPI接口读取2个以上的ADC芯片数据。

连续转换模式下,强烈建议采取上述两种方式之一。

11.1.5 低增益时的相关配置

当ADC增益选择为小于等于16倍时,需将SYS_CONF1里的GAIN_MODE置'1',大于16倍时为默认的'0'。

11.1.6 SYS_CONFx的几种典型配置值(仅基础配置,如有其它功能,参照手册描述进行相应配置)

11.2 不使用内部基准源的应用(例如电桥传感器应用)

如设置的资料率DR在400Hz以下时:

SYS_CONF0=0x00400000;SYS_CONF1=0x00010210;SYS_CONF2=0x00020000;

如设置的数据率DR在800Hz~6.4kHz时

SYS_CONF0=0x00400000;SYS_CONF1=0x00010210;SYS_CONF2=0x00000000;

如设置的数据率DR在12.8kHz以上时

SYS_CONF0=0x00400000;SYS_CONF1=0x00010210;SYS_CONF2=0x00030000;

如设置的数据率DR为51.2kHz时

SYS_CONF0=0x00400000;SYS_CONF1=0x00010230;SYS_CONF2=0x00030000;

11.3 使用内部基准源的应用

在上述桥式传感器各配置的基础上,将SYS_CONF1里的VRS位设置为1,SYS_CONF2里的BGP MODE位设置为1,REFTRIM<2:0>设置为'101'即可。

11.4 低功耗配置

在上述各配置的基础上,将SYS_CONF2的ADCKDIV2设置为1,再将SYS_CONF1的ADITA<1:0>设置为01,以及IDT<1:0>设置为10。

上述寄存器之外:

D_TARG在做系统增益校正时才需要使用。

12. 参考 DEMO 程序

```

unsigned long ADvalue;
unsigned char RH_DATA=0, RM_DATA=0, RL_DATA=0, RS_DATA=0;
unsigned long AD_H, AD_M, AD_L, AD_S;
unsigned char Rdata;
/*****
void SEND_ONE_BYTE(unsigned char COMMAND)
{ unsigned char i;
  for (i=8;i>0;i--) {if ((COMMAND&0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; COMMAND<<=1;}
}
/*****
void WRITE_REGISTER(unsigned char COMMAND, unsigned char H_DATA, unsigned char M_DATA, unsigned char L_DATA, unsigned char S_DATA)
{ unsigned char i;
  CS555X_SCLK=0; CS555X_CS=0; delay(5);
  for (i=8;i>0;i--) {if ((COMMAND&0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; COMMAND<<=1;}
  for (i=8;i>0;i--) {if ((H_DATA &0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; H_DATA <<=1;}
  for (i=8;i>0;i--) {if ((M_DATA &0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; M_DATA <<=1;}
  for (i=8;i>0;i--) {if ((L_DATA &0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; L_DATA <<=1;}
  for (i=8;i>0;i--) {if ((S_DATA &0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; S_DATA <<=1;}
  CS555X_CS=1; delay(5);}
/*****
unsigned char ReceiveByte(void)
{ unsigned char i;
  unsigned char dat=0;
  for (i=0;i<8;i++) {CS555X_SCLK=1; delay(1); dat<<=1; if (CS555X_SDO=1) {dat++;}; CS555X_SCLK=0; delay(1);}
  return dat;}
/*****
void READ_REGISTER(unsigned char COMMAND)
{ unsigned char i;
  CS555X_SCLK=0; CS555X_CS=0; delay(5);
  for (i=8;i>0;i--) {if ((COMMAND&0x80)==0) {CS555X_SDI=0;}else {CS555X_SDI=1;} delay(1); CS555X_SCLK=1; delay(1); CS555X_SCLK=0; COMMAND<<=1;}
  RH_DATA=ReceiveByte();
  RM_DATA=ReceiveByte();
  RL_DATA=ReceiveByte();
  RS_DATA=ReceiveByte();
  CS555X_CS=1; delay(5);}
/*****
void RESET_SPI(void)
{ CS555X_CS=0; delay(5);
  SEND_ONE_BYTE(0x00);
  SEND_ONE_BYTE(0xA5);
  SEND_ONE_BYTE(0xFF);
  SEND_ONE_BYTE(0x5A);
  CS555X_CS=1; delay(5); }
/*****
void INIT_CS555X(void)
{ CS555X_CS=1; delay(100);
  RESET_SPI();
  CS555X_CS=1; delay(100);
  WRITE_REGISTER(0x30, 0x80, 0x00, 0x00, 0x00); // RESET
  CS555X_CS=1; delay(100);
  WRITE_REGISTER(0x00, 0x00, 0x00, 0x00, 0x00); // OS_CHO
  WRITE_REGISTER(0x09, 0x02, 0x00, 0x00, 0x00); // GAIN_CHO
  WRITE_REGISTER(0x21, 0x00, 0x00, 0x00, 0x00); // CONV_CONF0 //PGA=128 DATA RATE=12.5Hz
  WRITE_REGISTER(0x28, 0x00, 0x00, 0x00, 0x00); // CONV_CONF1
  WRITE_REGISTER(0x30, 0x00, 0x01, 0x00, 0x00); // SYS_CONF0 //*****HBF_EN=0;
  WRITE_REGISTER(0x39, 0x00, 0x11, 0x02, 0x10); // SYS_CONF1 //*****FR_SEL=0;
  WRITE_REGISTER(0x41, 0x00, 0x02, 0x00, 0x00); // SYS_CONF2
  CS555X_CS=0; delay(5); SEND_ONE_BYTE(0x82); delay(1);
  return;}
/*****
void GET_ADC_CHO(void)
{ ADvalue=0;
  SEND_ONE_BYTE(0xFF); delay(1);
  AD_H=ReceiveByte();
  AD_M=ReceiveByte();
  AD_L=ReceiveByte();
  AD_S=ReceiveByte();
  ADvalue=ADvalue+AD_H; ADvalue=ADvalue<<8;
  ADvalue=ADvalue+AD_M; ADvalue=ADvalue<<8;
  ADvalue=ADvalue+AD_L; ADvalue=ADvalue<<8;
  ADvalue=ADvalue+AD_S;}
/*****
void READ_SET_DATA(void)
{ READ_REGISTER(0x24);
  LIShowNum(RS_DATA, 0);}

```