

单片机与 ADC0832 的接口技术

郑州铁路职业技术学院 梁明亮
中原工学院 王新强

ADC0832 是 NS (National Semiconductor) 公司生产的具有 Microwire/Plus 串行接口的 8 位 A/D 转换器,通过三线接口与单片机连接,功耗低,性能价格比较高,芯片引脚少,适宜在袖珍式智能仪器中使用。主要特点有:8 位分辨率,逐次逼近型,基准电压为 5V;输入模拟信号电压范围为 0~5V;输入和输出电平与 TTL 和 CMOS 兼容;在 250kHz 时钟频率时,转换时间为 32us;具有两个可供选择的模拟输入通道;功耗低,15mW。

一、引脚排列

ADC0832 有 DIP 和 SOIC 两种封装,DIP 封装的 ADC0832 引脚排列如图 1 所示。

各引脚说明如下:

CS 为片选端,低电平有效。

CH0,CH1 为两路模拟信号输入端。

DI 为两路模拟输入选择输入端。

DO 为模数转换结果串行输出端。

CLK 为串行时钟输入端。

VCC/REF 为正电源端和基准电压输入端。

GND 为电源地。

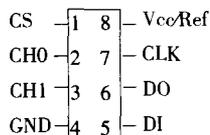


图 1 ADC0832 引脚图

二、配置位说明

ADC0832 工作时,模拟通道的选择及单端输入和差分输入的选择,都取决于输入时序的配置位。当差分输入时,要分配输入通道的极性,两个输入通道的任何一个通道都可作为正极或负极。ADC0832 的配置位逻辑表如表 1 所列。

表 1

输入格式	配置位		选择通道号	
	CH0	CH1	CH0	CH1
差分	L	L	+	-
	L	H	-	+
单端	H	L	+	
	H	H		+

表中“+”表示输入通道的端点为正极性;“-”表示输入端点为负极性;H 或 L 表示高、低电平。输入配置位时,高位(CH0)在前,低位(CH1)在后。

三、工作时序

当 CS 由高变低时,选中 ADC0832。在时钟的上升沿,DI 端的数据移入 ADC0832 内部的多路地址移位寄存器。在第一个时钟期间,DI 为高,表示启动位,紧接着输入两位配置位。当输入启动位和配置位后,选通输入模拟通道,转换开始。转换开始后,经过一个时钟周期延迟,以使选定的通道稳定。ADC0832 接着在第 4 个时钟下降沿输出转换数据。数据输出时先输出最高位(D7~D0);输出完转换结果后,又以最低位开始重新输出一遍数据(D7~D0),两次发送的最低位共用。当片选 CS 为高时,内部所有寄存器清 0,输出变为高阻态。如果要再进行一次模数转换,片选 CS 必须再次从高向低跳变,后面再输入启动位和配置位。

四、单片机与 ADC0832 的 SPI 串行典型接口

SPI (Serial Peripheral Interface) 是 MOTOROLA 公司推出的一种同步串行外设接口,允许 MCU 与各个厂家生产的标准外围设备接口,以串行方式交换信息。图 2 为 AT89S51 与 ADC 0832 的 SPI 串行接口方式,将 ADC 0832 的 CS 和 CLK 分别接单片机的 P1.4 和 P1.5 引脚,将 DI 和 DO 分别接 P1.6 和 P1.6 引脚。对 CH0 通道的模拟信号实行 A/D 转换,转换结果传送给 P0 口。下面以单片机汇编语言对其编程。

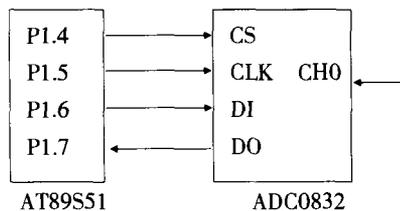


图 2: AT89S51 与 ADC0832 的 SPI 串行接口方式

```
CS EQU P1.4;片选
CLK EQU P1.5;串行时钟输入端
DI EQU P1.6;模拟输入选择输入端
DO EQU P1.7;数字量串行输出线
ORG0000H
START:CLR CS
MOV A,#03H;CH1=0,CH0=1,启动位=1,对 CH0 通道
```

网络拥塞控制中的公平性

信阳职业技术学院 苏锡亮 李 君

众所周知,网络拥塞控制一般要达到两个目标,即效率和公平。在现实生活中,对于公用资源的公平利用,其重要性甚至要超过对资源的有效使用。一个公用的网络应当尽量为每个用户提供公平的服务。

在讨论公平性之前,应先建立一个网络模型,即把网络看作一组链路的集合 L , 每条链路 $l(l \in L)$ 都有带宽 $C_l > 0$, 另外网络上还有一组业务流相互竞争以访问这些链路, 每个业务流与一条路径相关联, 一条路径是链路的集合 L 的一个子集。记 $l \in r$ 表示路径 r 穿过链路 l , 记 R 表示路径的集合。这里我们假设业务流是固定的, 研究公平性的目的就是寻找链路带宽的分配方法以使所有的流满足某种共享目标。记 λ_r 为路径 r 的带宽分配。很显然, 可行的带宽分配必须满足 $\sum_{r(l \in r)} \lambda_r \leq C_l$, $l \in L$ 的约束。

为讨论几种公平性, 我们用图 1 所示的一个简单的线性网络来阐述不同的分配策略。

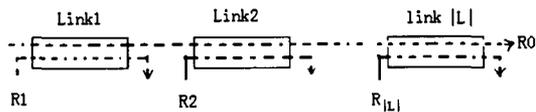


图 1 简单的线性网络模型

图中,网络由具有单位带宽容量的 $|L|$ 条链路组成, 有

x_0 条路径穿过所有链路, 这些路径的集合记为 R_0 ; 有 x_l 条路径仅仅穿过链路 l , 这些路径的集合记为 $R_l(l \in L)$ 。

要使网络达到最高使用效率的目标^[1], 也就对应着对带宽的可行分配 λ_r 应当满足最大化 $\sum_{r \in R} \lambda_r$ 的约束条件。然而这样的带宽分配目标却可能导致有些流根本分配不到带宽, 而这样做却是不公平的, 所以要想带宽分配达到公平性, 效率通常是要做折衷的, 也就是说, 带宽分配只能在满足某种公平性标准的基础上追求效率。

一、最大最小公平性

最大最小公平性最早出现在文献^[1]中, 用于定义数据网络公平性的标准。它的目标就是在网络容量的约束条件下, 可行流的分配 $\{\lambda_r, r \in R\}$ 中将最小的那一个可行分配最大化, 这也是它名称的由来。更确切地说, 最大最小公平性的可行分配可定义如下, 增加任何一个可行分配 $\lambda_{r'}$, 都必然导致某个其他的可行分配 λ_r ($\lambda_r < \lambda_{r'}$) 的降低。该定义的形式化表述为: 对每一条可行路径 r , 至少有一条链路 $l \in r$, 使(1)成立(已证明):

$$\sum_{r(l \in r)} \lambda_r = C_l \quad \text{且} \quad \lambda_r = \max\{\lambda_{r'}, l \in r'\} \quad (1)$$

最大最小公平性可以通过一个所谓的“填充过程”来实现。按照最大最小公平性的定义, 在图 1 的线性网络中, 业务流带宽分配应当如下式所示:

的模拟信号进行转换

MOV R7, #03H; 控制往 DI 写位的位数为 3 位, 即写 1 位启动位和 2 位配置位

L1: CLR CLK; CLK=0

RRC A

MOV DI, C; 写入一位

NOP;

SETB CLK; CLK=1

DJNZ R7, L1; 控制连续写三个位

CLR CLK; 延时一个时钟, 使选定的通道稳定

NOP;

SETB CLK

MOV R7, #08H; 控制读 DO 的位数为 8 位, 即读转换结果

L2: CLR CLK; CLK=0

MOV C, DO; 读入一位

RLC A; 转换结果暂存 A 中

SETB CLK; CLK=1

DJNZ R7, L2

SETB CS; 数字量串行输出结束, 置位 CS

MOV P0, A; 转换结果传送给 P0 口

SJMP START; 进行下一次 AD 转换

参考文献

[1] 王幸之, 等. AT89 系列单片机原理与接口技术. 北京: 北京航空航天大学出版社, 2004.

[2] 赖麒文. 8051 单片机 C 语言彻底应用. 北京: 科学技术出版社, 2002.