

Application Notes

3PA1030

3PA1030 是一款 10Bit、32MSPS 的流水线技术的模数转换器，其内部结构适合视频应用，也是为视频光传输领域特别设计的一款芯片，与 THS1030 及 AD9200 管脚兼容，且拥有更好的性能参数。

因为其内部结构的不同，在视频光传输应用时，在做参考电压，输入方式、范围，耦合模式以及钳位方式等各方面应用设计时需要注意某些不同点。

参考电压

3PA1030 的参考电压可方便的设为 0.5V 或者 1V。如果用户需要产生 0.5V 和 1V 以外的参考电压，可以使用外部电阻分压电路连接在 VREF，REFSENSE 和 GND 之间来产生 0.5V 与 1V 电压之间的任何参考电压。如图 1 所示。

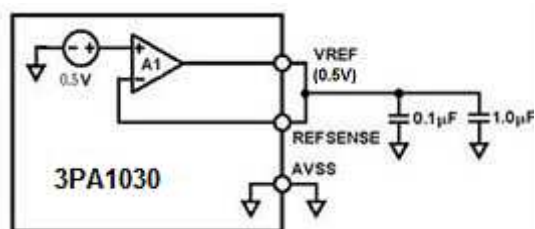


图 1a 0.5 V Reference(1V Input Span)

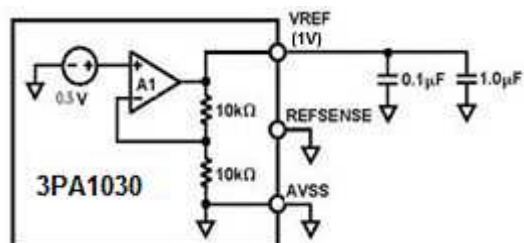


图 1b 1V Reference(2V Input Span)

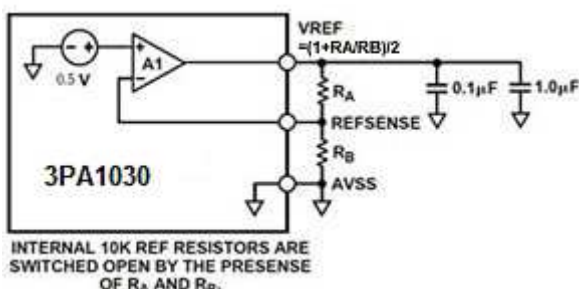


图 1c Variable Reference(Between 0.5 V and 1 V)

当视频滤波的输出电压不在 0~2 伏时，需调整视频滤波放大倍数，来匹配 3PA1030 的输入电压。

钳位电路

3PA1030 有内部钳位电路。用户可使用 3PA1030 内部钳位电路将输入模拟信号的 DC 电平钳位在需要的电压，如图 2 所示。

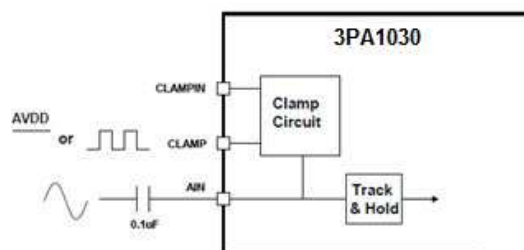


图 2a Bottom signal level clamped to CLAMPIN

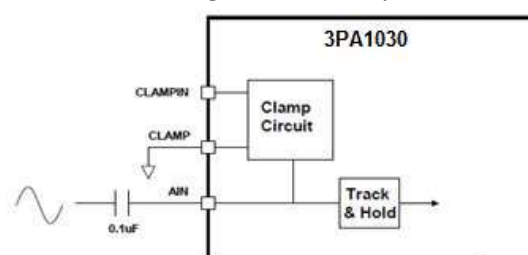


图 2b Bottom signal level clamped to ground

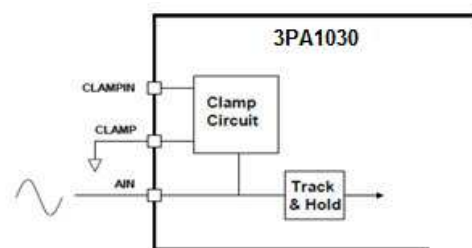


图 2c DC-coupled input

当输入范围在 0~2 伏时，推荐使用直流耦合。

输入范围

3PA1030 为固定的参考电压，其输入范围为 0~2 伏。如果视频滤波的输入范围不在 0~2 伏时，需调整放大倍数，来匹配 3PA1030 的输入范围，以期达到更好的动态特性。

耦合方式

直流耦合或交流耦合，3PA1030 都可以正常工作。当视频滤波输出不匹配的时候，需要使用交流耦合。当视频滤波输出与 3PA1030 匹配时，推荐使用直流耦合。

3PD5651 / 3PD5651E

3PD5651/3PD5651E 为 10 比特, 125 兆转换率的开关电流型, 电压输出数模转换器, 适用于通信, 仪器仪表与视频重建等应用。在视频光端机应用中, 与 THS5651 及 AD9760 管脚兼容。但在内部参考, 输出范围等方面不同, 需要做一些调整, 得到最好的产品性能。

3PD5651 和 3PD5651E 的时钟时序相位相差 180 度, 客户可根据自己原有程序的时序选择使用 3PD5651 或者 3PD5651E 产品。

参考电压

3PD5651/3PD5651E 的内部参考电压由 FSADJ 输出为 1.1 伏。使用时需在 FSADJ 端和 GND 直接接电阻 Rset。

$IREF = V_{FSADJ} / R_{set}$

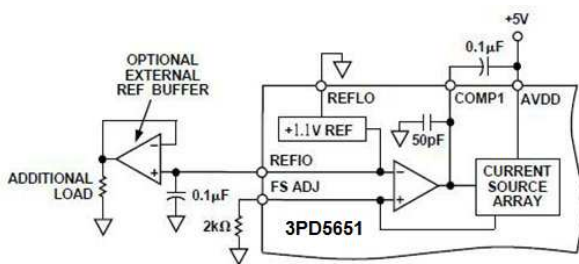


图 1 3PD5651/3PD5651E 内部参考电压

输出类型及范围

3PD5651/3PD5651E 的输出为双端互补电流输出 loutA 和 loutB。输出范围 $I_{outFS} = 32 \times I_{REF}$ 。可通过在 loutA 和 loutB 端接电阻负载 Rload 将电流输出转换为需要的电压输出。 $V_{out} = I_{out} \times R_{set}$ 。当视频滤波的输入范围与其不匹配的时候, 需要交流耦合。视频滤波的放大倍率根据实际情况做相应调整。当视频滤波的输入范围与其匹配时, 可以直流耦合。

Comp 接法

Comp 脚为内部基准去耦管脚。使用时, COMP1 通过 0.1μF 电容接 AVDD, COMP2 通过 0.1μF 电容接 AGND。

时序调整

下面的 FPGA (或 CPLD) 参考程序用来说明时序

调整方法。该程序是一个传送 4 路 ADC 数据的 VHDL 范例程序, 建议大家使用 3PA1030 时参照范例程序中的时序编写程序。

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
entity send is
    port(clk : in std_logic;
          data1,data2,data3,data4:in std_logic_vector(9 downto 0);
          data_out :out std_logic_vector(9 downto 0);
          clk1,clk2,clk3,clk4 : out std_logic
    );
end send;
architecture demo_architecture of send is
    signal cnt: std_logic_vector(1 downto 0);
    signal dbuf1,dbuf2,dbuf3,dbuf4:std_logic_vector(9 downto 0);
    begin
        process(clk)
            begin
                if clk'event and clk='1' then
                    cnt<=cnt+1;
                    case cnt is
                        when "00"=>
                            data_out(9 downto 0)<=dbuf1;
                        when "01"=>
                            dbuf1<=data1;
                            dbuf2<=data2;
                            dbuf3<=data3;
                            dbuf4<=data4;
                            data_out(9 downto 0)<=dbuf2;
                        when "10"=>
                            data_out(9 downto 0)<=dbuf3;
                        when "11"=>
                            data_out(9 downto 0)<=dbuf4;
                        when others=>
                            data_out(9 downto 0)<=(others=>'Z');
                            --null;
                    end case;
                end if;
            end process;

            clk1<= cnt(1);
            clk2<= cnt(1);
            clk3<= cnt(1);
            clk4<= cnt(1);
        end demo_architecture;
    
```

该程序在一个时刻同时读取 4 路 ADC 数据, 放到 FPGA (或 CPLD) 的内部寄存器里, 再用一个 4 倍频时钟把寄存器里的 4 路 ADC 数据从一组数字端口顺序送出。由于 4 路 ADC 的采样时钟统一, 为了容易调节时序, 4 路 ADC 数据应该在同一时刻读入寄存器。

唯一对 ADC 时序有影响的是具体哪个时刻读取 ADC 数据。如果在使用 ADC 的过程中, 某个或某几个通道会有雪花点出现, 应该是这些通道的 ADC

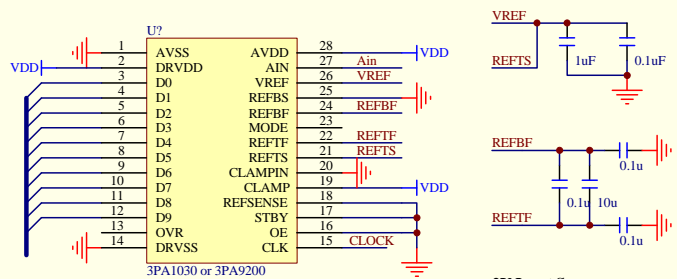
的时序不太匹配，此时可以通过下面调整解决问题：
将下面四句程序的位置做下调整。

```
dbuf1<=data1;  
dbuf2<=data2;  
dbuf3<=data3;  
dbuf4<=data4;
```

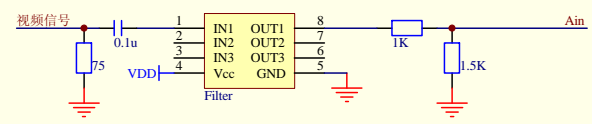
比如对应出现雪花的 ADC 通道的那句程序本来是放在 cnt=00 时的，便移到 cnt=10 时的，此时相当于相位反转了 180 度，此时采样便不会出现问题了。图像出现雪花点的问题便可解决。

不同厂家的芯片在时序上会有一些差异。思瑞浦公司对光端机上的常用品牌的 ADC 做过对比测试，不同品牌间的时序大概差异不超过 10ns，而光端机应用中一个 ADC 时钟周期在 60ns 以上，所以按照上面方法，对出现问题的通道时序翻转 180 度，修改后的程序基本可以匹配市面上的 ADC 芯片。

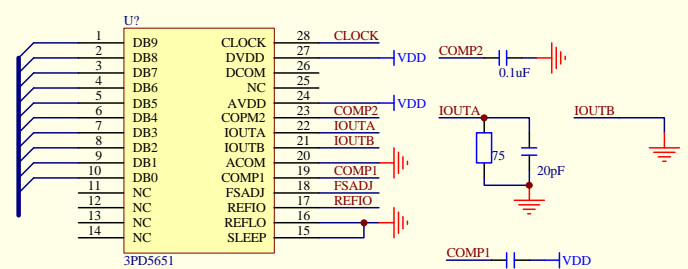
3PD5651 或者 3PD5651E 的时序调整方法原则上类似。由于一般应用中多路 DAC 使用统一时钟，从 FPGA 或 CPLD 送到多路 DAC 的数据应该在同一时刻发送。



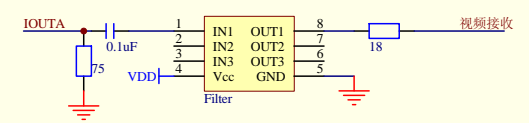
2V Input Span



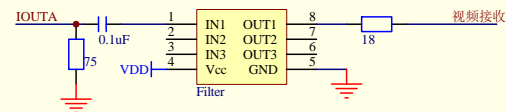
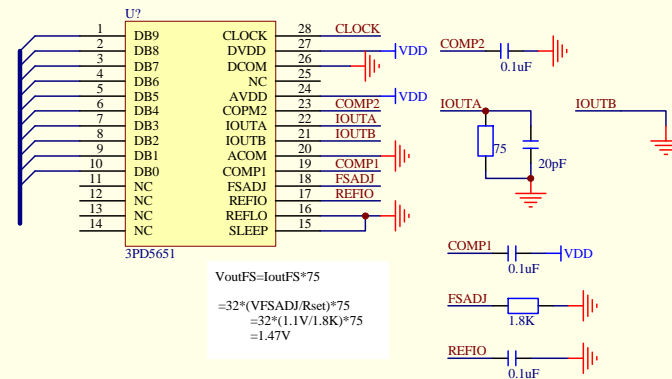
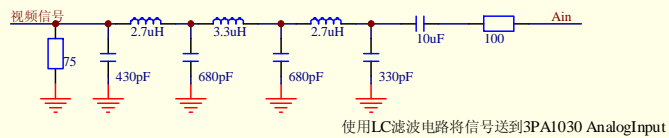
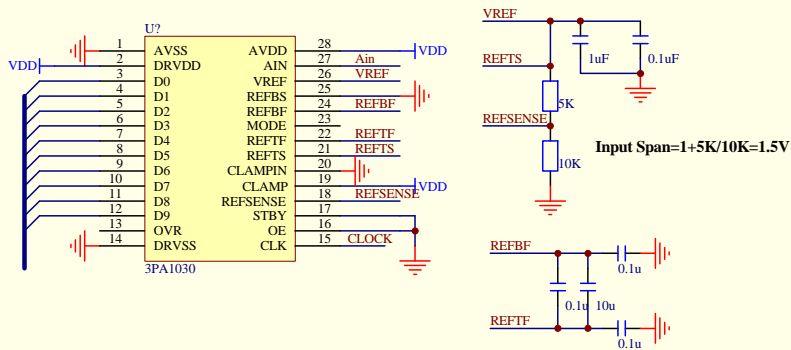
使用有源滤波器将信号送到3PA1030 AnalogInput



$$\begin{aligned}
 V_{outFS} &= I_{outFS} * 75 \\
 &= 32 * (V_{FSADJ} / R_{set}) * 75 \\
 &= 32 * (1.1V / 1.8K) * 75 \\
 &= 1.47V
 \end{aligned}$$



Title		
Size	Number	Revision
A4		
Date:	27-Jun-2011	Sheet of
File:	F:\AD DA 应用\应用\应用.ddb	Drawn By:



Title		
Size A4	Number	Revision
Date: 5-Aug-2011	Sheet of	
File: E:\AD DA 应用\应用\应用.dtb	Drawn By:	