



<http://www.Microcontrol.cn> 微控设计网

中国 MSP430 单片机专业网站

MSP430F2 系列 16 位超低功耗单片机模块原理

第 14 章 OA 运算放大器

版本： 1.4

日期： 2007.5.

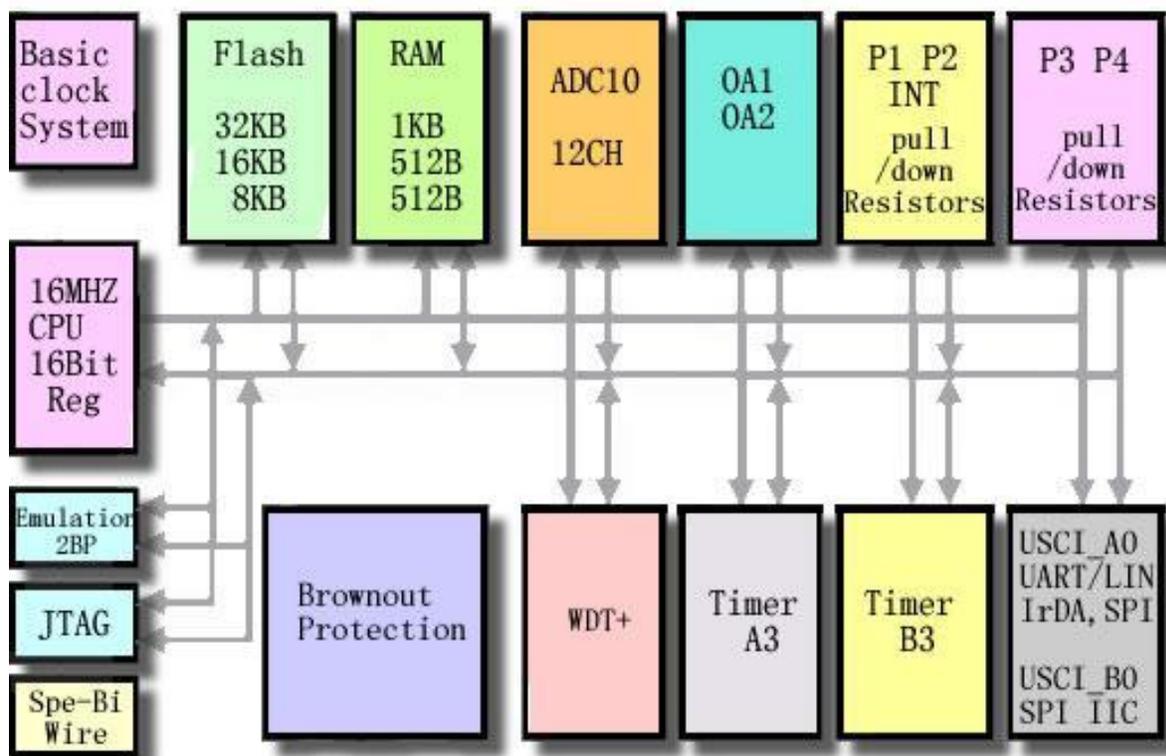
原文： TI MSP430x2xxfamily.pdf

翻译： 陈军 长春工业大学

编辑： DC 微控技术论坛总版主

注： 以下文章是翻译 TI MSP430x2xxfamily.pdf 文件中的部分内容。由于我们翻译水平有限，有整理过程中难免有所不足或错误；所以以下内容只供参考。一切以原文为准。

详情请密切留意微控技术论坛。



第 14 章 OA

OA 是通用运算放大器。本章描述了 OA。在 MSP430x22x4 配置了两个 OA 模块。

14.1 OA 简介

OA 可用于在 AD 转换前的模拟信号调理。

OA 特征如下：

- | 单电源供电，工作电流低；
- | 轨对轨输出；
- | 可通过程序设置稳定时间；
- | 可通过软件选择配置模式；
- | 可通过软件选择反馈电阻，以应用在 PGA（比例增益放大器）上。

注意：多 OA 模块

有些芯片内置了一个以上的 OA 模块。在这种情况下，各 OA 模块独立工作。在本节里，会出现如 OA_xCTLO 这样的用来描述寄存器名字的术语。这时，x 常用来表示讨论的是哪个 OA 模块。独立工作时，寄存器像 OA_xCTLO 这样简写。

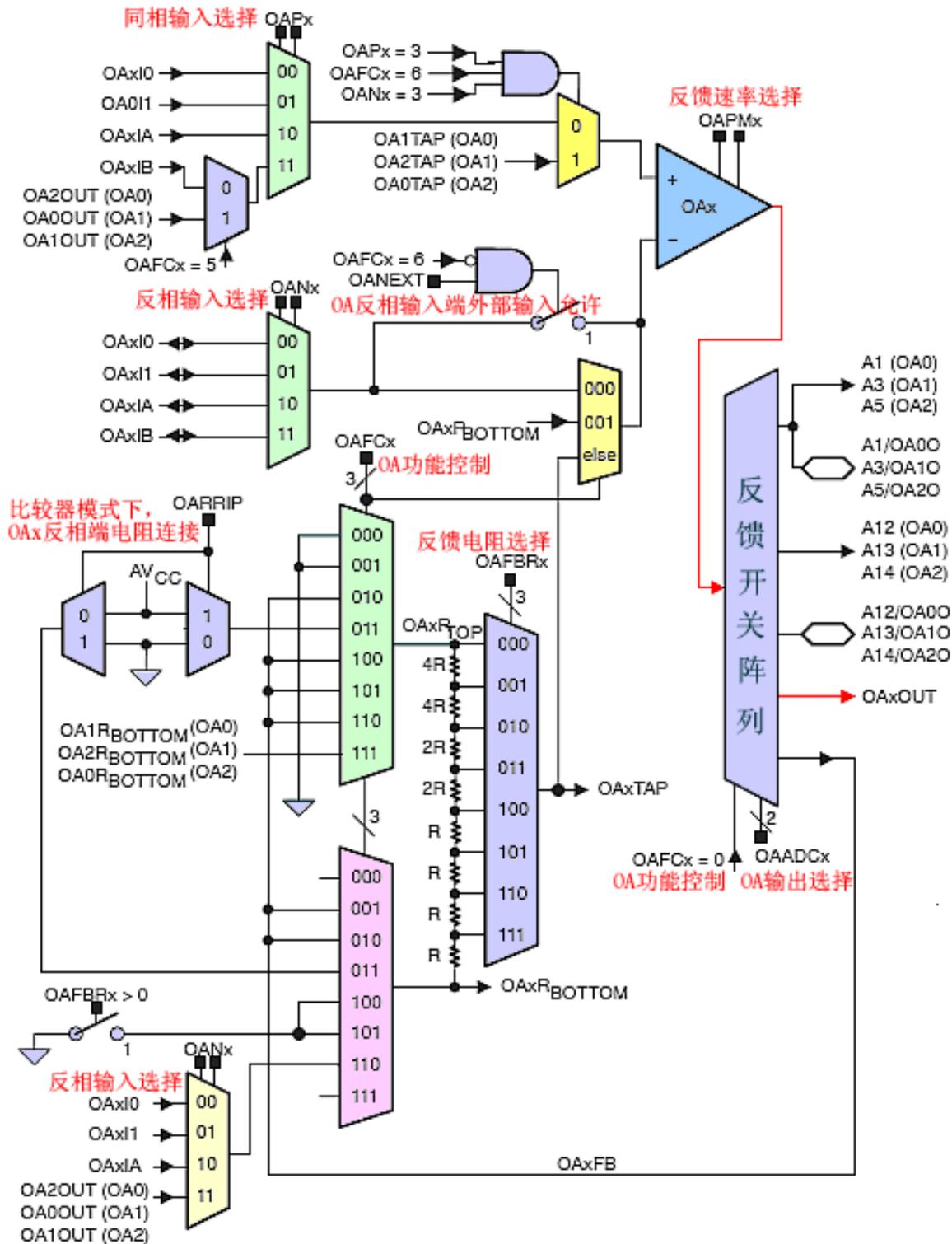


图 14-1 OA 模块框图

14.2 OA 操作

OA 模块由用户通过软件设置。下面部分将讨论 OA 的设置和操作。

14.2.1 OA 放大器

OA 是可配置、低电流、轨对轨输出的运算放大器。它可以被配置成反相放大器，或同相放大器，或与其它 OA 模块组合形成差分放大器。OA 的输出反馈速率可以用 OAPMx 位配置成最佳的稳定时间。当 OAPMx = 00 时，OA 关闭，输出为高阻态。当 OAPMx > 0 时，OA 打开。参数见具体的模块数据手册。

14.2.2 OA 输入

OA 具有可配置的输入选择。+和-端输入可以通过 OANx and OAPx 位进行独立选择，可以选择为外部信号或内部信号。OA0I1 提供一个同相输入，在所有 OA 模块的内部都有同相输入。OAxIA 和 OAxIB 提供依赖器件的输入。信号连接参考器件数据手册。当外部反相输入不需要时，置位 OANEXT 位，可以使内部反相输入用于外部。

14.2.3 OA 输出和反馈线路

OA 具有可配置的输出选择，由 OAADCx 位和 OAFcx 位控制。OA 输出信号可以在内部供给 ADC10 输入的 A12(OA0)，A13 (OA1) 和 A14 (OA2)，或者连接到这些 ADC 的输入和其外部引脚。OA 输出也可以通过设置 OAFcx 位连接到一个内部梯形电阻网络。梯形电阻网络抽头可由 OAFBRx 位选择，具有可编程增益放大功能。

表 14-1 表明了 OA 输出和反馈线路配置。当 OAFcx = 0 时，OA 为通用模式，器件的反馈在外部完成。当 OAFcx > 0 和 OAADCx = 00 或 11 时，OA 的输出通过内部路径连接到器件。当 OAFcx > 0 和 OAADCx = 01 或 10，OA 输出由内部和外部路径连接到器件。

表 14-1 OA 输出配置

OAFcx	OAADCx	OA 输出和反馈线路
= 0	x0	OAxOUT 连接到外部引脚和 ADC 输入的 A1, A3 或 A5.
= 0	x1	OAxOUT 连接到外部引脚和 ADC 输入的 A12, A13 或 A14.
> 0	00	OAxOUT 只用作内部线路.
> 0	01	OAxOUT 连接到外部引脚和 ADC 输入的 A12, A13 或 A1
> 0	10	OAxOUT 连接到外部引脚和 ADC 输入的 A1, A3 或 A5.
> 0	11	OAxOUT 通过内部连接到 ADC 输入的 A12, A13 或 A14。外部 A12, A13 或 A14 引脚与 ADC 连接断开。

14.2.4 OA 配置

OA 可以通过 OAFcx 位配置具有不同的放大功能，见表 14-2。

表 14-2 OA 模式选择

OAFcx	OA 模式
000	通用运算放大器
001	用于三运放差分放大器的单位增益缓冲器
010	单位缓冲器（相当于电压跟随器）
011	比较器
100	同相 PGA(比例放大)放大器
101	级连同相 PGA 放大器
110	反相 PGA 放大器
111	差分放大器

通用运算放大器模式

在该模式下，反馈梯形电阻网络与 OA_x 是独立的，由 OA_xCTL0 位定义信号路径。 OA_x 输入由 OAP_x 和 OAN_x 位选择。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

用于差分放大器的单位增益模式

该模式下， OA_x 的输出连接到 OA_x 的反相输入端， OA_x 提供一个单位增益缓冲器。同相输入由 OAP_x 位选择。连接到反相输入端的外部连接禁止， OAN_x 位无需关注（状态任意）。

OA_x 的输出连接到梯形电阻网络，作为三运放差分放大器的一部分。该模式只用于三运放差分放大器结构。

单位增益模式

该模式下， OA_x 的输出连接到 OA_x 的反相输入端， OA_x 提供一个单位增益缓冲器（也称为电压跟随器）。同相输入由 OAP_x 位选择。连接到反相输入端的外部连接禁止， OAN_x 位无需关注。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

比较器模式

该模式下，反馈梯形电阻网络与 OA_x 是独立的。当 $OARRIP = 0$ 时， $RTOP$ 连接到 AVSS， $RBOTTOM$ 连接到 AVCC。当 $OARRIP = 1$ 时，梯形电阻网络的连接反相， $RTOP$ 连接到 AVCC， $RBOTTOM$ 连接到 AVSS。 OA_xTAP 信号连接到 OA_x 的反相输入端， OA_x 提供一个具有可编程门限电压的比较器，该电压由 $OAFBR_x$ 位确定。同相输入端由 OAP_x 位确定。通过外部反相反馈电阻可以增加延迟。反相输入端的外部连接被禁止， OAN_x 位无需关注。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

同相 PGA 模式

该模式下， OA_x 的输出连接到 $RTOP$ ， $RBOTTOM$ 连接到 AVSS。 OA_xTAP 连接到 OA_x 的反相输入端。 OA_x 提供一个具有 $[1+OA_xTAP \text{ 比值}]$ 增益的可编程同相放大器。 OA_xTAP 比值由 $OAFBR_x$ 选择，如果 $OAFBR_x = 0$ ，增益是 1。同相输入由 OAP_x 位选择。反相输入引脚上的外部连接被禁止， OAN_x 位无需关注（状态任意）。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

级联同相 PGA 模式

该模式允许 OA 信号在内部按照反相模式级联 2 个或 3 个 OA 。该模式下，当 $OAP_x = 11$ 时， OA_x 的反相输入连接到 $OA20UT$ ($OA0$)， $OA00UT$ ($OA1$) 或 $OA10UT$ ($OA2$)。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

反相 PGA 模式

该模式下， OA_x 的输出连接到 $RTOP$ ， $RBOTTOM$ 连接到一个模拟多路复用器，它可以复用 OA_xI0 ， OA_xI1 ， OA_xIA 或由 OAN_x 位选择的剩余的一个 OA_s 的输出。 OA_xTAP 信号连接到 OA_x 的反相输入端。 OA_x 提供一个具有 $-OA_xTAP$ 比值增益的反相放大器。 OA_xTAP 比值由 $OAFBR_x$ 位选择。 OAP_x 位确定同相输入。 OA_x 输出可以通过 OA_xCTL0 位来选择连接到 ADC10 输入通道。

注意：使用 OA_x 负输入同时用作 ADC 输入

当引脚连接到反相输入多路复用器时，也被用作了 ADC 的输入，由于内部线路上的压降，转换结果的误差可能上升到 5mV。

差分放大器模式

该模式允许 OA 信号在内部连接形成一个两运放或三运放的仪表放大器。图 14-2 表示了 $OA0$ 和 $OA1$ 构成的两运放配置。该模式下，通过经由反相 PGA 模式下的另一 OA_x ， OA_x 的输出连接到

RTOP。RBOTTOM 断开，提供一个单位增益缓冲器。该缓冲器与 1 到 2 个剩下的 OAx 组合形成差分放大器。OAx 输出可以通过 OAxCTL0 位来选择连接到 ADC10 输入通道。图 14-2 表示了一个由 OA0 和 OA1 构成的两运放差分放大器。控制寄存器设置见表 14-3。放大器的增益由 OA1 的 OAFBRx 位决定，见表 14-4。OAx 的相互连接见图 14-3。

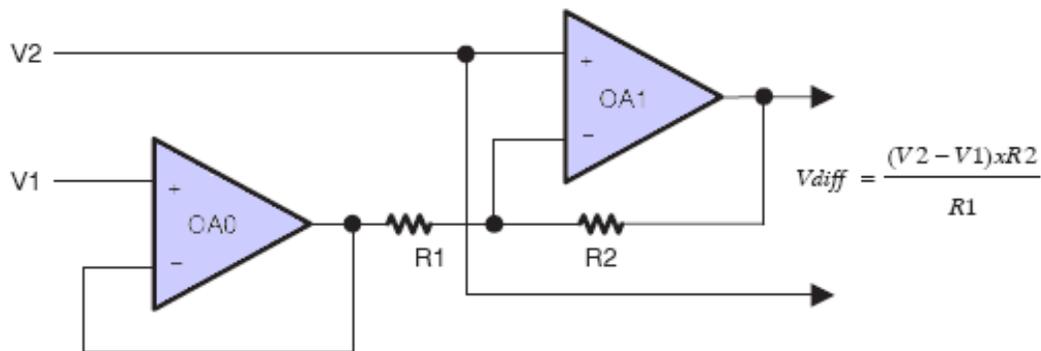
表 14-3 两运放差分放大器控制寄存器设置

寄存器	设置(二进制)
OA0CTL0	xx xx xx 0 0
OA0CTL1	000 111 0 x

表 14-3 两运放差分放大器增益设置

OA1 OAFBRx	Gain
000	0
001	1/3
010	1
011	1 2/3
100	3
101	4 1/3
110	7
111	15

图 14-2 两运放差分放大器



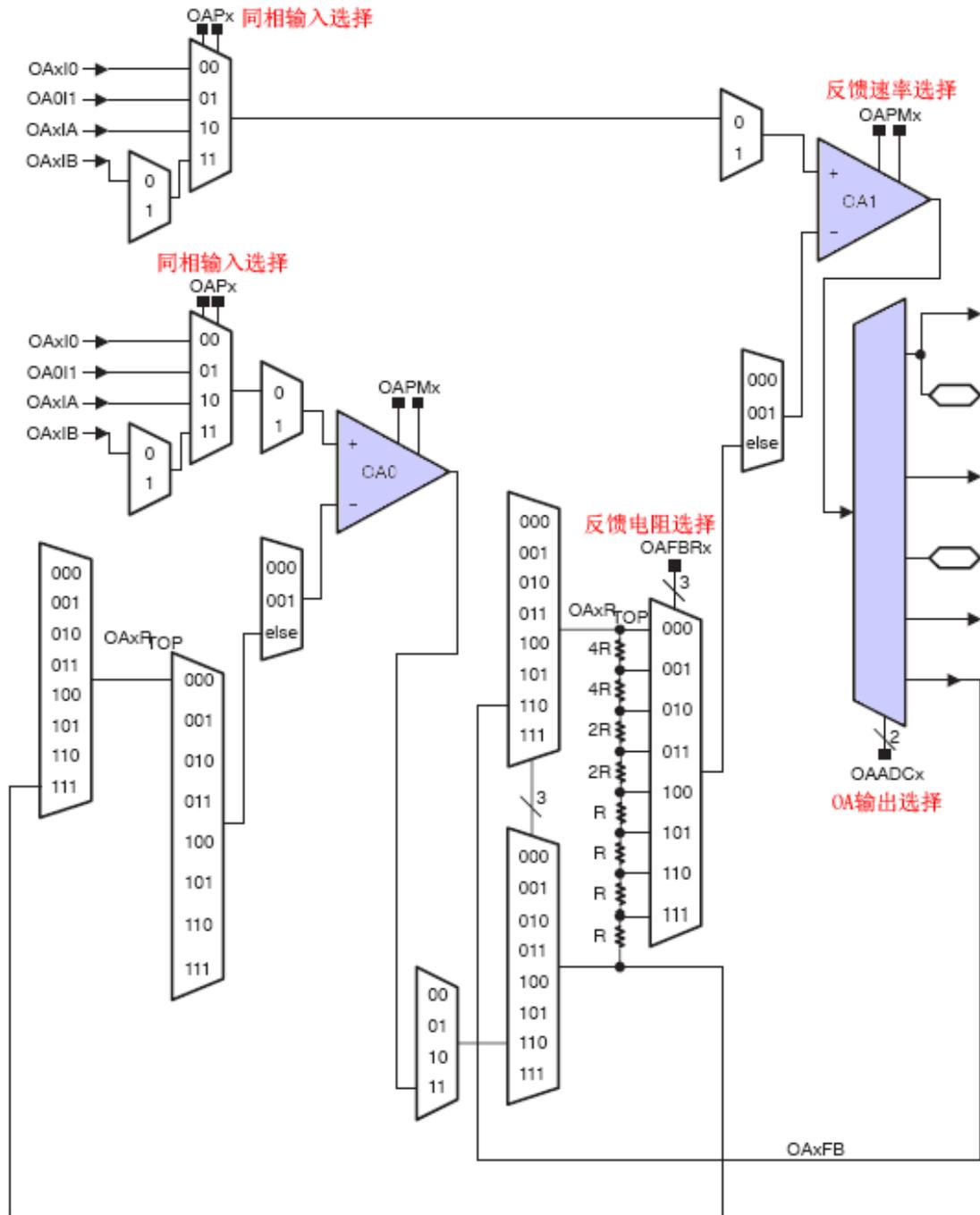


图 14-3 两运放差分放大器 OAx 相互连接

图 14-4 表示了使用 OA0, OA1, OA2 构成的三运放差分放大器。(三运放不是所有的器件中都具有, 参见具体的器件数据手册)。控制寄存器的设置见表 14-5。其增益由 OA0, OA2 的 OAFBRx 位确定。OA0, OA2 的 OAFBRx 位的设定必须相同。增益设置见表 14-6。OAx 相互连接见表 14-5。

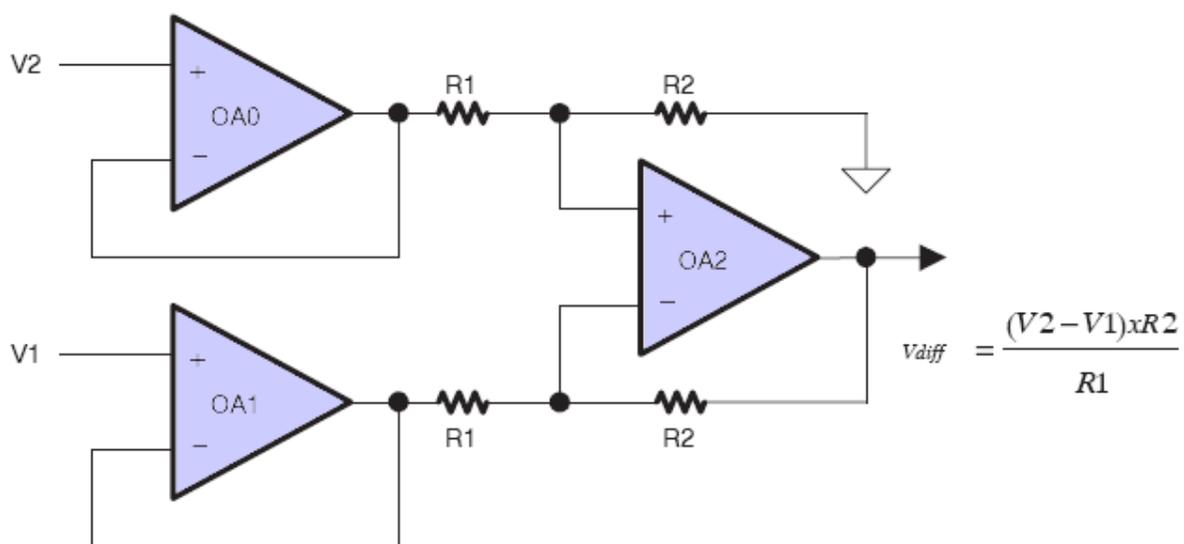
表 14-5 三运放差分放大器控制寄存器设置

寄存器	设置(二进制)
OA0CTL0	xx xx xx 0 0
OA0CTL1	xxx 001 0 x
OA1CTL0	xx xx xx 0 0

表 14-6 三运放差分放大器增益设置

OA0/OA2 OAFBRx	Gain
000	0
001	1/3
010	1
011	1 2/3
100	3
101	4 1/3
110	7
111	15

图 14-4 三运放差分放大器



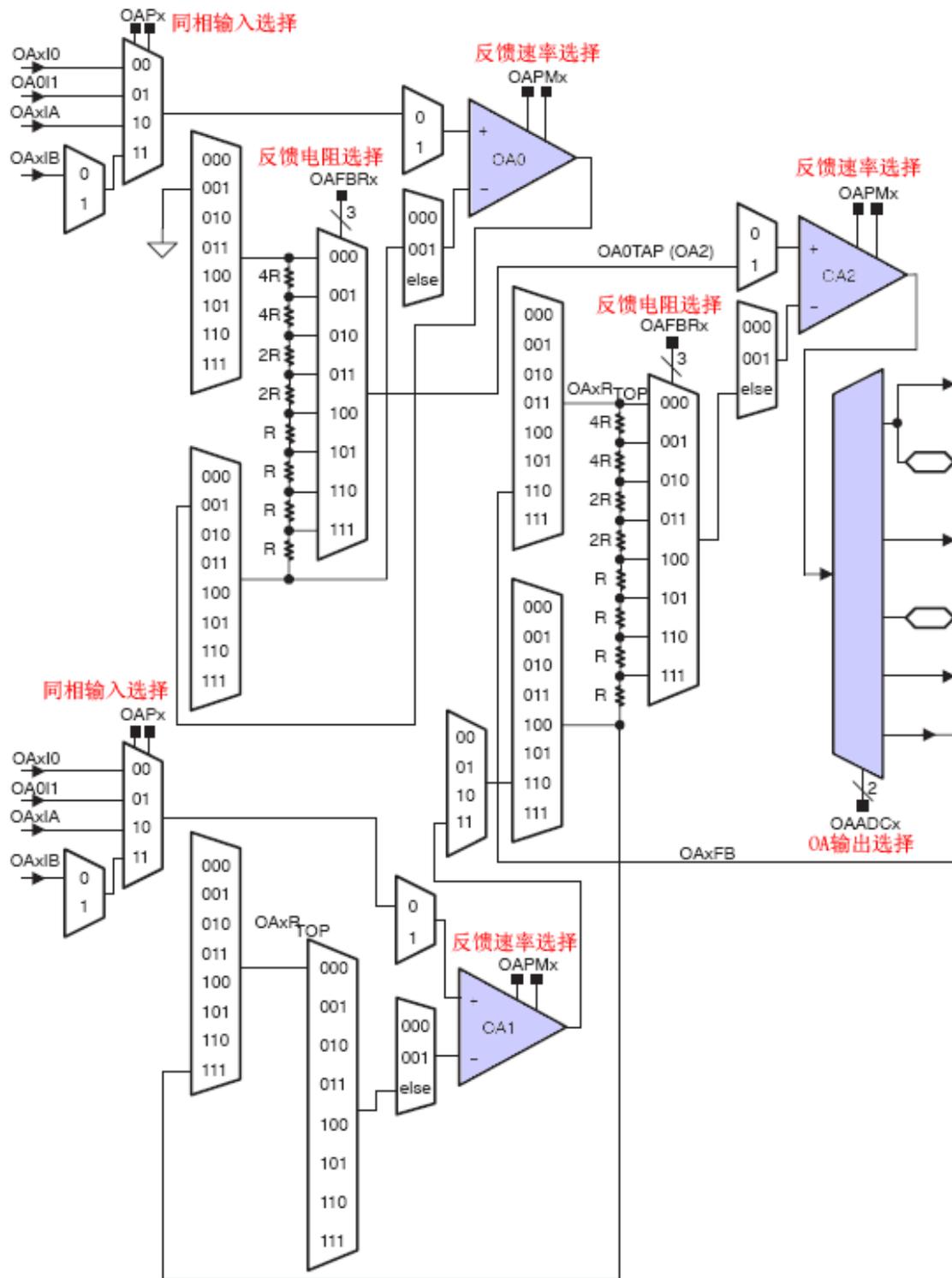


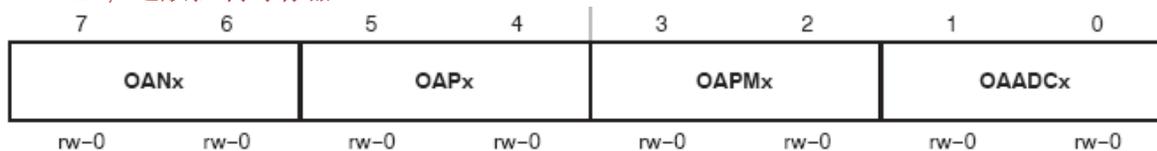
图 14-5 三运放差分放大器 OAx 相互连接

14.3 OA 寄存器

寄存器	格式	寄存器类型	地址	初始化状态
OA0控制寄存器0	OA0CTL0	读/写	0C0h	复位和POR
OA0控制寄存器1	OA0CTL1	读/写	0C1h	复位和POR

OA1 控制寄存器0	OA1CTL0	读/写	0C2h	复位和 POR
OA1控制寄存器1	OA1CTL1	读/写	0C3h	复位和 POR
OA2控制寄存器0	OA2CTL0	读/写	0C4h	复位和 POR
OA2控制寄存器1	OA2CTL1	读/写	0C5h	复位和 POR

OA_xCTL0, 运放控制寄存器 0



OAN_x Bits7-6 反相输入选择。这两位用于选择 OA 反相输入端的输入信号。

- 00 OA_xI0
- 01 OA_xI1
- 10 OA_xIA—连接信号参考数据手册
- 11 OA_xIB—连接信号参考数据手册

OAP_x Bits5-4 同相输入选择。这两位用于选择 OA 同相输入端的输入信号。

- 00 OA_xI0
- 01 OA_xO11
- 10 OA_xIA—连接信号参考数据手册
- 11 OA_xIB—连接信号参考数据手册

OAPM_x Bits3-2 反馈速率选择。

- 00 关闭, 输出高阻态
- 01 低速
- 10 中速
- 11 快速

OAADC_x Bits1-0 OA 输出选择。这两位和 OAFC_x 位一起控制 OAPM_x > 0 时 OA_x 的输出路径。

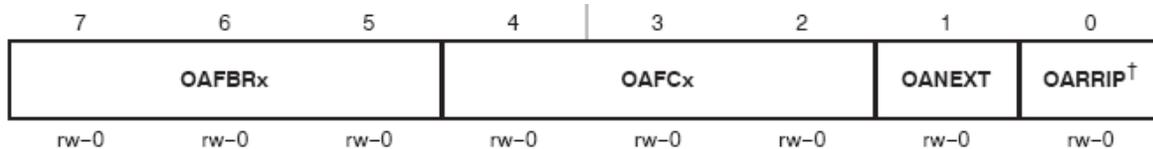
当 OAFC_x = 0:

- 00 OA_xOUT 连接到外部引脚和 ADC 输入的 A1, A3 或 A5
- 01 OA_xOUT 连接到外部引脚和 ADC 输入的 A12, A13 或 A14
- 10 OA_xOUT 连接到外部引脚和 ADC 输入的 A1, A3 或 A5
- 11 OA_xOUT 连接到外部引脚和 ADC 输入的 A12, A13 或 A14

当 OAFC_x > 0:

- 00 OA_xOUT 只用在内部连接。
- 01 OA_xOUT 连接到外部引脚和 ADC 输入的 A12, A13 或 A14
- 10 OA_xOUT 连接到外部引脚和 ADC 输入的 A1, A3 或 A5
- 11 OA_xOUT 内部连接到 ADC 输入的 A12, A13 或 A14. 外部 A12, A13 或 A14 与 ADC 断开连接。

OA_xCTL1, 运放控制寄存器 1



OAFBRx Bits 7-5 OAx 反馈电阻选择。

- 000 抽头 0 — 0R/16R
- 001 抽头 1 — 4R/12R
- 010 抽头 2 — 8R/8R
- 011 抽头 3 — 10R/6R
- 100 抽头 4 — 12R/4R
- 101 抽头 5 — 13R/3R
- 110 抽头 6 — 14R/2R
- 111 抽头 7 — 15R/1R

O AFCx Bits 4-2 OAx 功能控制。选择 OAx 的功能。

- 000 通用运算放大器
- 001 用于三运放差分放大器的单位增益缓冲器
- 010 单位增益缓冲器（相当于电压跟随器）
- 011 比较器
- 100 同相 PGA 放大器
- 101 级联同相 PGA 放大器
- 110 反相 PGA 放大器
- 111 差分放大器

OANEXT Bit 1 OAx 反相输入端外部输入允许。当该位被置位且使用内部集成电阻网络时，将 OAx 反相输入连接到外部引脚。

- 0 OAx 反相输入端不能从外部输入
- 1 OAx 反相输入端可以从外部输入

OARRIP Bit 0 比较器模式下，OAx 反相端电阻连接。

- 0 当 O AFCx = 3，RTOP 连接到 AVSS，RBOTTOM 连接 AVCC。
- 1 当 O AFCx = 3，RTOP 连接到 AVCC，RBOTTOM 连接 AVSS。

OA 基础例程

[1] OA0 工作在模式 2 (单一增益放大)

```

//*****
//p2.0 为正端输入
//p2.1 输出(最大输 AVCC)
//将输入电压的地接到芯片的 AVss
//用 OAFBR 来选择反馈电阻值
//将直流电源正端接入 p2.0,调节电压值,观参输出是否跟输入保持一致
//
//          MSP430F2274
//          -----
//          /\|          XIN|-
//          | |          |
//          --|RST      XOUT|-

```

```

//          |          |
//          |          | P2.0|<--OA0IO
//          |          | AVss|<--GND
//          |          | p2.1|-->OA00
//
#include "msp430x22x4.h"
int main( void )
{
    WDTCTL=WDTPW+WDTHOLD;           //关看门狗
    BCSCTL1 =CALBC1_1MHZ;           //设定 DCO 为 1MHZ
    DCOCTL =CALBC1_1MHZ;
    P2SEL |=BIT0+BIT1;              //P2.0,2.1 第二功能选择
    P2OUT |=BIT1;                    //p2.1,OA0 输出端(不加也没关系)
    OA0CTL0=OAP_0+OAPM_3+OAADC1;    //OAP 选择 OA0IO 输入,反馈速度为 fast,输出为 A1,A3 或者 A5
    OA0CTL1=OAFBR_1+OAFRC_2;        //R2/R1=4R/16R,反馈电阻为 4R
    //OA0CTL1=OAFBR_2+OAFRC_2;      //R2/R1=8R/8R,反馈电阻为 8R
    //OA0CTL1=OAFBR_3+OAFRC_2;      //R2/R1=10R/6R,反馈电阻为 10R
    //OA0CTL1=OAFBR_4+OAFRC_2;      //R2/R1=12R/4R,反馈电阻为 12R
    //OA0CTL1=OAFBR_5+OAFRC_2;      //R2/R1=13R/3R,反馈电阻为 13R
    //OA0CTL1=OAFBR_6+OAFRC_2;      //R2/R1=14R/2R,反馈电阻为 14R
    //OA0CTL1=OAFBR_7+OAFRC_2;      //R2/R1=15R/1R,反馈电阻为 15R
    _NOP();
    _BIS_SR(LPM3_bits);             //进入低功耗 3
    _NOP();
    return 0;
}
//*****

```

[2] OA0 工作在模式 3(比较器)

```

//*****
//OA0 工作在模式 3(comparator)
//p2.0 为正端输入(0~Vcc)
//将输入电压的地接到芯片的 AVss
//用 OAFBR 来选择反向端参考电压
//将直流电源接入 p2.0,调节电压值,观参输出端的变化

//          MSP430F2274
//          -----
//          /|\          XIN|-
//          ||          |
//          --|RST      XOUT|-
//          |          AVcc|<--DVCC
//          |          P2.0|<--OA0IO
//          |          AVss|<--GND
//          |          p2.1|-->OA00

```

```

#include "msp430x22x4.h"
int main( void )
{
    WDTCTL=WDTPW+WDTHOLD;           //关看门狗
    BCSCTL1 =CALBC1_1MHZ;           //设定 DCO 为 1MHZ
    DCOCTL =CALBC1_1MHZ;
    P2SEL |=BIT0+BIT1;              //选择功能模块功能

```

```

P2OUT |=BIT1; //P21 为输出
OA0CTL0=OAP_0+OAPM_3+OAADC1; //OAP 选择 OA010 输入, 反馈速度为 fast, 输出为 A1,A3 或者 A5
OA0CTL1=OAFBR_1+OAF_3; //运放反向端输入为 4/16AVcc; 比较器模式
//OA0CTL1=OAFBR_2+OAF_3; //运放反向端输入为 8/16AVcc; 比较器模式
//OA0CTL1=OAFBR_3+OAF_3; //运放反向端输入为 10/16AVcc; 比较器模式
//OA0CTL1=OAFBR_4+OAF_3; //运放反向端输入为 12/16AVcc; 比较器模式
//OA0CTL1=OAFBR_5+OAF_3; //运放反向端输入为 13/16AVcc; 比较器模式
//OA0CTL1=OAFBR_6+OAF_3; //运放反向端输入为 14/16AVcc; 比较器模式
//OA0CTL1=OAFBR_7+OAF_3; //运放反向端输入为 16/16AVcc; 比较器模式
_NOP();
_BIS_SR(LPM3_bits); //进入低功耗 3
_NOP();
return 0;
}
//*****

```

[3] OA0 工作在模式 4(同相比例放大)

```

//*****
//p2.0 输入(零点几个伏特)
//p2.1 输出(最大输出 AVCC)
//将输入电压的地接到芯片的 AVss
//用万用表来观察在各个放大系数时输出端电压
//
//          MSP430F2274
//
//          -----
//          /|\|          XIN|-
//          | |          |
//          --|RST       XOUT|-
//          |           |
//          |           P2.0|<--OA010
//          |           AVss|<--GND
//          |           p2.1|-->OA00
#include "msp430x22x4.h"
int main( void )
{
    WDTCTL=WDTPW+WDTHOLD; //关看门狗
    BCSCCTL1 =CALBC1_1MHZ; //设定 DCO 为 1MHZ
    DCOCTL =CALBC1_1MHZ;
    P2SEL |=BIT0+BIT1; //P2.0,2.1 第二功能选择
    P2OUT |=BIT1; //p2.1,OA0 输出端(不加也没关系)
    OA0CTL0=OAP_0+OAPM_3+OAADC1; //OAP 选择 OA010 输入, 反馈速度为 fast, 输出为 A1,A3 或者 A5
    OA0CTL1=OAFBR_0+OAF_4; //R2/R1=0R/16R, 模式 4(同向比例放大器, 放大系数=1+R2/R1)
    OA0CTL1=OAFBR_1+OAF_4; //R2/R1=4R/12R
    OA0CTL1=OAFBR_2+OAF_4; //R2/R1=8R/8R
    OA0CTL1=OAFBR_3+OAF_4; //R2/R1=10R/6R
    OA0CTL1=OAFBR_4+OAF_4; //R2/R1=12R/4R
    OA0CTL1=OAFBR_5+OAF_4; //R2/R1=13R/3R
    OA0CTL1=OAFBR_6+OAF_4; //R2/R1=14R/2R
    OA0CTL1=OAFBR_7+OAF_4; //R2/R1=15R/1R
    _NOP();
    _BIS_SR(LPM3_bits); //进入低功耗 3
    _NOP();
    return 0;
}
//*****

```



```
ADC10CTL1 = INCH_3; //选择第 3 通道
ADC10CTL0 = REFON+SREF_1+REF2_5V; //打开 2.5V 正参考,地为负参考
ADC10CTL0 | = ADC10ON+ADC10SHT_3+ADC10IE;
//打开 ADC10 内核,设定采样保持时间为 64 个 ADC10CLK,使能 ADC10 中断
ADC10CTL0 | = ENC+ADC10SC; //启动 AD 转换
_BIS_SR(GIE); //使能总中断
_NOP();
return 0;
}

#pragma vector = ADC10_VECTOR
__interrupt void adc_isr (void)
{
    adc_value = ADC10MEM; //将 AD 值存入 adc_value
    _NOP();
}
//*****
```



MSP430F22x4 开发板

专业提供 MSP430 单片机开发工具