



介绍

本应用手册适用于需要 ACM32G103 系列芯片运算放大器（OPA）模块的使用者。它描述了与运算放大器模块相关的设置和功能使用方法，以便在应用程序中进行优化设计。

本应用说明应与相关的用户手册、数据表一同阅读。

1. 概述

OPA 即 OPAMP 模块包含三个独立配置的运算放大器。每个运算放大器有两个输入和一个输出，三个 I/O 可以连接到外部引脚(SA 模式)。运算放大器可以在内部配置为一个跟随器(UG 模式)，或者是一个可编程增益的放大器 (PGA 模式)。

具有以下特性：

- 轨到轨输入输出；
- 可配置成运放外置模式；
- 可配置成 PGA 模式，增益可编程；
- 可配置成跟随器模式，实现单位增益；
- 运算放大器输出到管脚；
- 运算放大器输出可作为 ADC 的输入；
- 运算放大器正端输入来自管脚或者是 DAC 的输出；
- 运算放大器负端输入来自管脚、UG 反馈或者是 PGA 的反馈电阻网络输出。

2. 功能描述

2.1. 输入输出选择

标准运放输入输出包含 VIN+、VIN-和 VOUT，分别对应本运放模块中的 VINP、VINM 和 VOUT。本运放模块可以通过控制寄存器配置输入和输出，非常灵活方便。

- 正端输入选择

正端输入 VINP 可以通过配置控制寄存器 OPAX_CSR 的 VINP_SEL 位域来选择不同输入，输入可来自：

- 外部管脚输入
- 芯片内部 DAC 输出
- 芯片内部 OPA 输出

- 负端输入选择

负端输入 VINM 可以通过结合运放工作模式和 VINM0_EN 来配置，输入可来自：

- 内部（PGA 正向放大和 UG 模式）
- 外部管脚输入（VINM0 或 VINM1）

- 输出选择

输出 VOUT 可以通过配置控制寄存器 OPAX_CSR 的 OUT_SEL 位域选择，可输出到：

- 外部管脚
- 芯片内部 ADC 输入端

2.2. 操作模式和校准

运放可工作在正常模式和校准模式。校准模式下软件可对运放的失调电压进行校准。

表格 2-1 运放工作模式控制和输出关系

Mode	Control bits			Output	
	EN	CAL NEN	CAL PEN	Vout	CALout flag
Normal	1	0	0	analog	0
		1	1	analog	0
Power down	0	x	x	z	0
Offset cal n	1	1	0	analog	x

Offset cal p	1	0	1	analog	x
--------------	---	---	---	--------	---

注：当 Mode 为 Normal 模式，EN=1，CAL_NEN=1，CAL_PEN=1，校准值生效。当 Mode 为 Normal 模式，EN=1，CAL_NEN=0，CAL_PEN=0，校准值无效，此时采用默认值。

2.3. 工作模式和负端输入

通过 MODE_SEL 设置运放的工作模式，可以设置为运放外置、单位增益和可编程增益放大模式。注意不同工作模式的负端输入选择，模式选择为 2'b00 和 2'b11 的两种运放外置模式的主要区别是负端输入的不同，需注意 VINM0_EN 按照对应模式正确配置。

表格 2-2 工作模式和负端输入

工作模式	MODE_SEL	VINM0_EN	VINM1	VINM0
SA(运放外置)	00	0	输入	-
UG(单位增益)	01	0	-	-
PGA(倍数可调)	10	0	反相输入	-
SA(运放外置)	11	1	-	输入

注：若需要使用 PGA 模式，则只能选择 VINM1 作为负端输入。

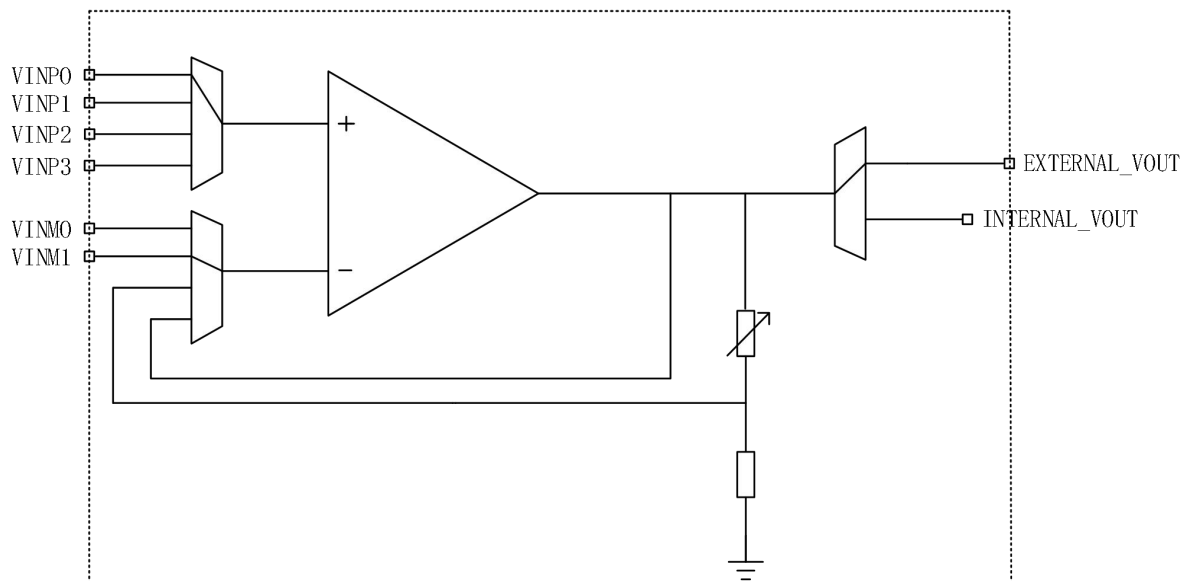
表格 2-3 负端输入管脚

OPAMP 编号	VINM1	VINM0
OPAMP1	PC5	PA3
OPAMP2	PC5	PA5
OPAMP3	PB10	PB2

2.4. 运放外置模式（SA 模式）

可通过配置寄存器中 MODE_SEL 位域将运放配置成外置模式，此时内部电阻被切断，需通过外部电路来调整运放功能。等效的电路如下图所示：

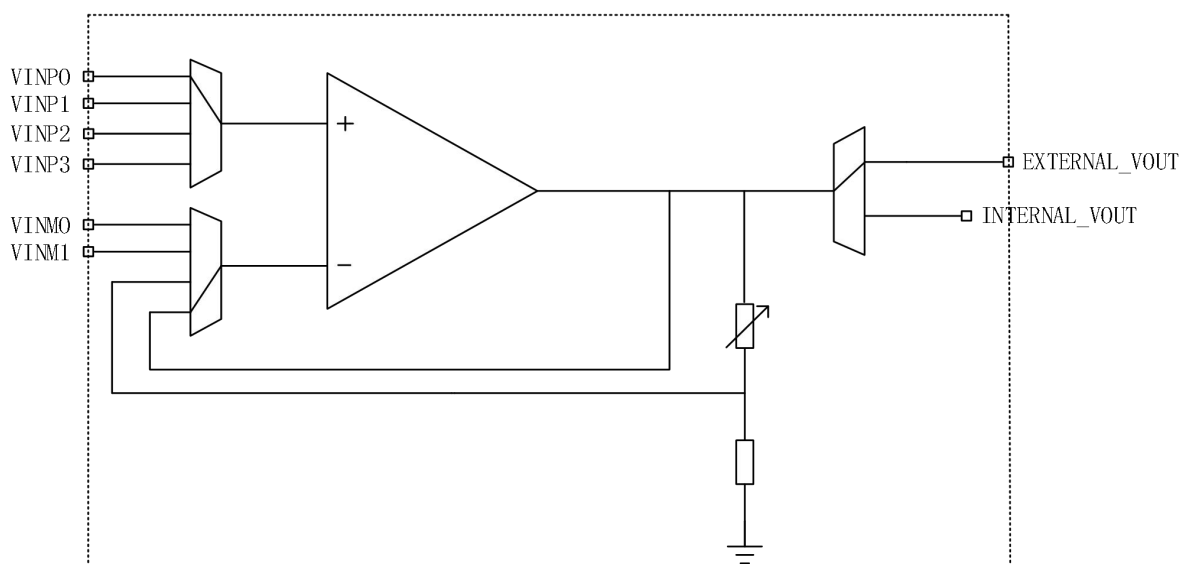
图 2-1 运放外置模式配置等效图



2.5. 运放单位增益模式（UG 模式）

可通过配置寄存器中 MODE_SEL 位域将运放配置成单位增益模式，此时内部电阻被切断，负端输入直接连接到输出端，此时运放增益为 1。等效的电路如下图所示：

图 2-2 运放单位增益模式配置等效图



2.6. 可编程增益放大模式（PGA 模式）

可通过配置寄存器中 MODE_SEL 位域将运放配置成可编程增益放大模式。可以通过寄存器配置运放的放大倍数，从而可省略外部电阻。配置 OPAx_CSR 中的 POL_SEL 位域可以配

置运放为同相或反相模式，配置 GAIN_SEL 位域可以编程增益倍数，同相和反相的增益有区别。

配置为 PGA 同相模式时，负端输入端无效；配置为 PGA 反相模式时，正端输入和负端输入均有效。

图 2-3 运放 PGA 同相模式配置等效图

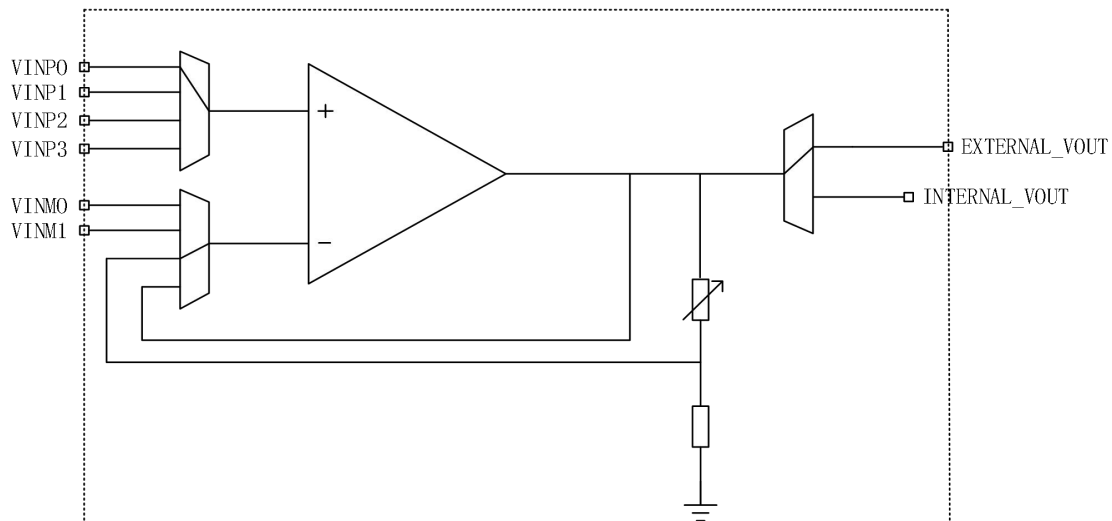
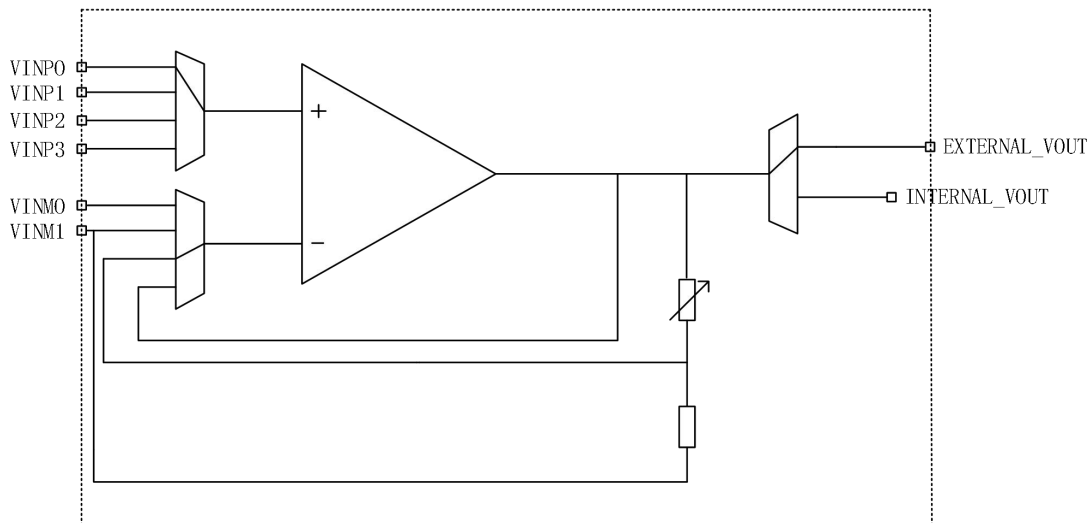


图 2-4 运放 PGA 反相模式配置等效图



3. 应用例程说明

3.1. 单位增益模式例程（OPA_UG）

3.1.1. 硬件层配置

在 HAL_OPA_MspInit 函数中，将需要用到的管脚设置为模拟端口，例程中包含了三颗运放的设置方法之一，这里代码需要用户根据具体应用情况自行设置修改。

例程中 VINP 指的是运放正端输入 VIN+，VINM 指的是负端输入 VIN-，VOUT 即是输出。

例程中 OPA2 的管脚设置，其中 VINM 设置给 SA 模式使用的，在 UG 和 PGA 模式中不需要使用。

例：

```
else if(hopa->Init.OPAx == OPA2)
{
    //OPA2 VinP0: PA7
    //OPA2 VinM1: PC5
    //OPA2 Vout: PA6

    __HAL_RCC_GPIOA_CLK_ENABLE();

    GPIO_Handle.Pin    = GPIO_PIN_6 | GPIO_PIN_7;
    GPIO_Handle.Mode   = GPIO_MODE_ANALOG;
    GPIO_Handle.Pull   = GPIO_NOPULL;
    GPIO_Handle.Drive  = GPIO_DRIVE_LEVEL3;
    HAL_GPIO_Init(GPIOA, &GPIO_Handle);

    __HAL_RCC_GPIOC_CLK_ENABLE();

    GPIO_Handle.Pin    = GPIO_PIN_5;
    GPIO_Handle.Mode   = GPIO_MODE_ANALOG;
    GPIO_Handle.Pull   = GPIO_NOPULL;
    GPIO_Handle.Drive  = GPIO_DRIVE_LEVEL3;
    HAL_GPIO_Init(GPIOC, &GPIO_Handle);
}
```

3.1.2. 初始化设置

- 首先定义 OPA 的结构体变量

例如：

```
OPA_HandleTypeDef OPA2_Handle;
```

- 然后给所需的结构体成员变量赋值

UG 模式配置如下：

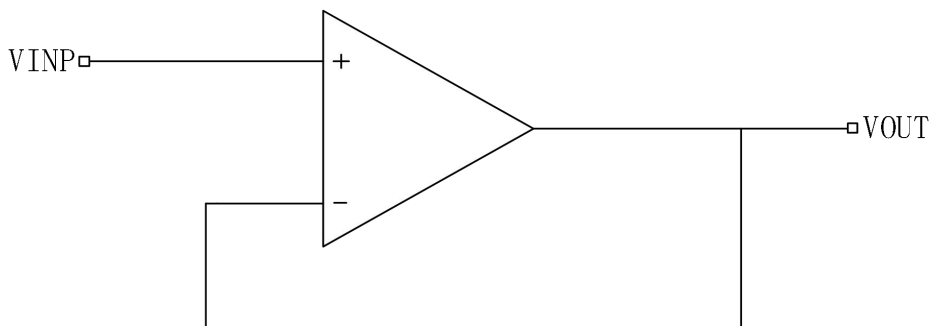
```
void OPA_Init_UG(void)
{
    OPA2_Handle.Instance = OPA;//赋值 OPA 基地址
    OPA2_Handle.Init.OPAx = OPA2;//指定需要配置的运放号
    OPA2_Handle.Init.Mode = OPA_MODE_UG;//设置为单位增益模式
    OPA2_Handle.Init.VinPSel = OPA_VINP_0;//设置正端输入选择为 VINP0（PC5）
    OPA2_Handle.Init.OutSel = OPA_OUT_GPIO;//比较结果输出到管脚
    OPA2_Handle.Init.Hsm = OPA_HSM_HIGH;//设置为高驱动模式
    OPA2_Handle.Init.TrimEn = OPA_TRIM_DISABLE;//不校准

    OPA2_Handle.Init.VinM0En = OPA_VINM0_DISABLE;//负端选择，只有 SA1(MODE_SEL=11b)模式使能
    OPA2_Handle.Init.Gain = OPA_GAIN_2_1;//设置增益倍数，只在 PGA 模式生效
    OPA2_Handle.Init.PolSel = OPA_POL_NOINVERT;//同相反相选择，只在 PGA 模式生效

    HAL_OPA_Init(&OPA2_Handle);//执行初始化
}
```

所有参数在运放驱动的.h 文件中都有定义，用户根据自己的电路来配置。完成初始化后，运放就可以使用了。

设置成 UG 模式后，运放电路等效如下图所示：



3.2. 可编程增益放大模式例程（OPA_PGA）

3.2.1. 硬件层配置

同 3.1.1.1. 章节。

3.2.2. 初始化设置

- 首先定义 OPA 的结构体变量

例如：

```
OPA_HandleTypeDef OPA2_Handle;
```

- 然后给所需的结构体成员变量赋值

PGA 模式配置如下：

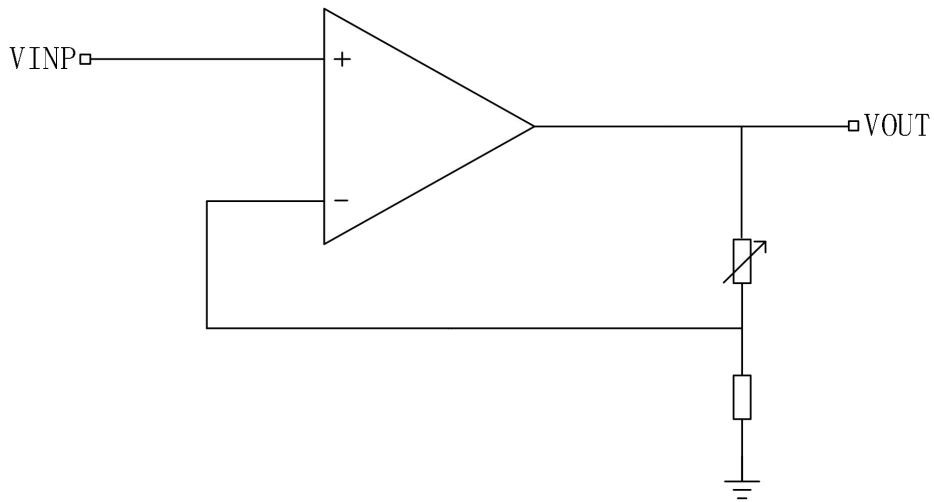
```
void OPA_Init_PGA(void)
{
    OPA2_Handle.Instance = OPA;//赋值 OPA 基地址
    OPA2_Handle.Init.OPAx = OPA2;//指定需要配置的运放号
    OPA2_Handle.Init.Mode = OPA_MODE_PGA; //设置为可编程增益放大模式
    OPA2_Handle.Init.VinPSel = OPA_VINP_0;//设置正端输入选择为 VINP0（PC5）
    OPA2_Handle.Init.OutSel = OPA_OUT_GPIO;//比较结果输出到管脚
    OPA2_Handle.Init.PolSel = OPA_POL_INVERT;//反相模式：正端负端输入都有效
    OPA2_Handle.Init.Hsm = OPA_HSM_HIGH;//设置为高驱动模式
    OPA2_Handle.Init.Gain = OPA_GAIN_2_1;//设置增益倍数，此处为正向放大 2 倍
    OPA2_Handle.Init.TrimEn = OPA_TRIM_DISABLE;//不校准

    OPA2_Handle.Init.VinM0En = OPA_VINM0_DISABLE; //负端选择，只有 SA1(MODE_SEL=11b)模式使能

    HAL_OPA_Init(&OPA2_Handle);
}
```

所有参数在运放驱动的.h 文件中都有定义，用户根据自己的电路来配置。完成初始化后，运放就可以使用了。

设置成 PGA 模式后，运放电路等效如下图所示：



3.3. 外置模式例程（OPA_SA）

3.3.1. 硬件层配置

同 3.1.1. 章节。

3.3.2. 初始化设置

- 首先定义 OPA 的结构体变量

例如：

```
OPA_HandleTypeDef OPA2_Handle;
```

- 然后给所需的结构体成员变量赋值

SA 模式如下：

```
void OPA_Init_SA(void)
{
```

```
    OPA2_Handle.Instance = OPA;//赋值 OPA 基地址
```

```
    OPA2_Handle.Init.OPAx = OPA2;//指定需要配置的运放号
```

```
    OPA2_Handle.Init.Mode = OPA_MODE_SA;//设置为外置模式 SA
```

```
    OPA2_Handle.Init.VinPSel = OPA_VINP_0;//设置正端输入选择为 VINP0（PC5）
```

```
    OPA2_Handle.Init.OutSel = OPA_OUT_GPIO;//比较结果输出到管脚
```

```
    OPA2_Handle.Init.Hsm = OPA_HSM_HIGH;//设置为高驱动模式
```

```
    OPA2_Handle.Init.TrimEn = OPA_TRIM_DISABLE;//不校准
```

```
    OPA2_Handle.Init.VinM0En = OPA_VINM0_DISABLE; //负端选择，只有 SA1(MODE_SEL=11b)模式使能
```

```
    OPA2_Handle.Init.Gain = OPA_GAIN_2_1;//设置增益倍数，只在 PGA 模式生效
```

```

OPA2_Handle.Init.PolSel = OPA_POL_NOINVERT;//同相反相选择，只在 PGA 模式生效

HAL_OPA_Init(&OPA2_Handle);//执行初始化
}
void OPA_Init_SA1(void)
{
    OPA2_Handle.Instance = OPA;//赋值 OPA 基地址
    OPA2_Handle.Init.OPAx = OPA2;//指定需要配置的运放号
    OPA2_Handle.Init.Mode = OPA_MODE_SA1;//设置为外置模式 SA1
    OPA2_Handle.Init.VinPSel = OPA_VINP_0;//设置正端输入选择为 VINP0（PC5）
    OPA2_Handle.Init.VinM0En = OPA_VINM0_ENABLE; //设置 VINM0 使能，负端选择 VINM0（PA5）
    OPA2_Handle.Init.OutSel = OPA_OUT_GPIO;//比较结果输出到管脚
    OPA2_Handle.Init.Hsm = OPA_HSM_HIGH;//设置为高驱动模式
    OPA2_Handle.Init.TrimEn = OPA_TRIM_DISABLE;//不校准

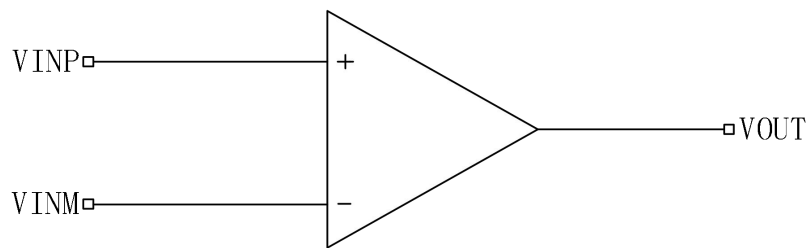
    OPA2_Handle.Init.PolSel = OPA_POL_NOINVERT;//同相反相选择，只在 PGA 模式生效
    OPA2_Handle.Init.Gain = OPA_GAIN_2_1;//设置增益倍数，只在 PGA 模式生效

    HAL_OPA_Init(&OPA2_Handle);//执行初始化
}

```

所有参数在运放驱动的.h 文件中都有定义，用户根据自己的电路来配置。完成初始化后，运放就可以使用了。

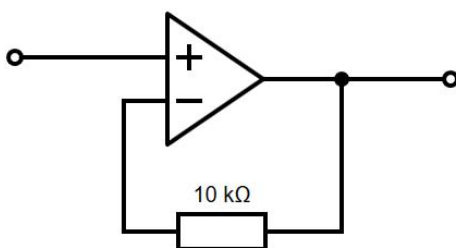
设置成 SA 模式后，运放所有引脚都连接到管脚，电路等效如下图所示：



4. 外置模式（SA）运放应用电路举例

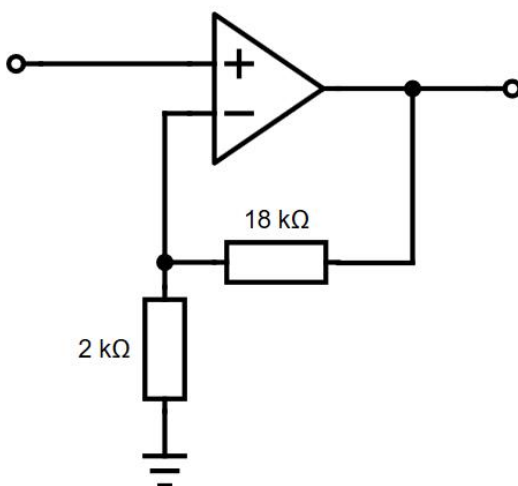
4.1. 单位增益缓冲电路

单位增益缓冲电路用于阻抗匹配，高输入阻抗，低输出阻抗，增益为 1。典型应用电路如下图所示：



4.2. 直流信号放大电路

下图是一个典型同相 10 倍直流放大电路的应用电路图。根据负反馈电阻和负反馈落地电阻计算电路的直流增益为： $A = 1 + R_f / R$ ；其中 $R_f = 18\text{k}\Omega$ ， $R = 2\text{k}\Omega$ ， $A = 10$ ；

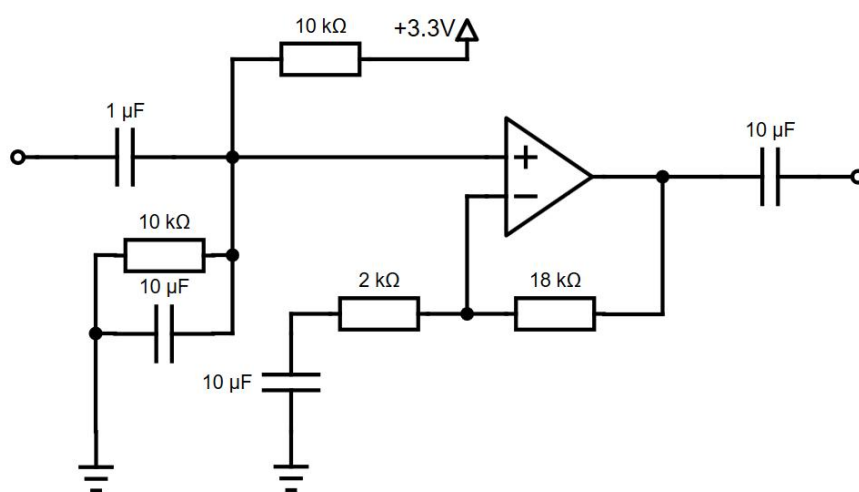


4.3. 交流信号放大电路

下图是一个典型同相 10 倍交流放大电路的应用电路图。电路的直流增益为 1，交流增益为： $A = 1 + R_f / R$ ；其中 $R_f = 18\text{k}\Omega$ ， $R = 2\text{k}\Omega$ ， $A = 10$ 。

需要注意的是，由于运放采用 MCU 的 VDDA 单电源供电，交流信号放大电路与直流放大电路有很大不同。主要有如下几点：

- 需要输入、输出隔直电容；
- 需要在正端输入提供 $VDDA/2$ 的中点偏置电压；
- 需要反馈落地隔直电容；
- 需要注意电容和阻抗形成的一阶 RC 滤波效应影响电路的带宽，一般-3dB 带宽截止频率点为 $F_c = 1 / (2 * \pi * R * C)$ 。



联系我们

公司：上海航芯电子科技有限公司
地址：上海市闵行区合川路 2570 号科技绿洲三期 2 号楼 702 室
邮编：200241
电话：+86-21-6125 9080
传真：+86-21-6125 9080-830
Email: Service@AisinoChip.com
Website: www.aisinochip.com

版本维护

版本	日期	作者	描述
V1.0	2023-05-09	Aisinochip	初始版

本文档的所有部分，其著作权归上海航芯电子科技有限公司（简称航芯公司）所有，未经航芯公司授权许可，任何个人及组织不得复制、转载、仿制本文档的全部或部分组件。本文档没有任何形式的担保、立场表达或其他暗示，若有任何因本文档或其中提及的产品所有资讯所引起的直接或间接损失，航芯公司及所属员工恕不为其担保任何责任。除此以外，本文档所提到的产品规格及资讯仅供参考，内容亦会随时更新，恕不另行通知。