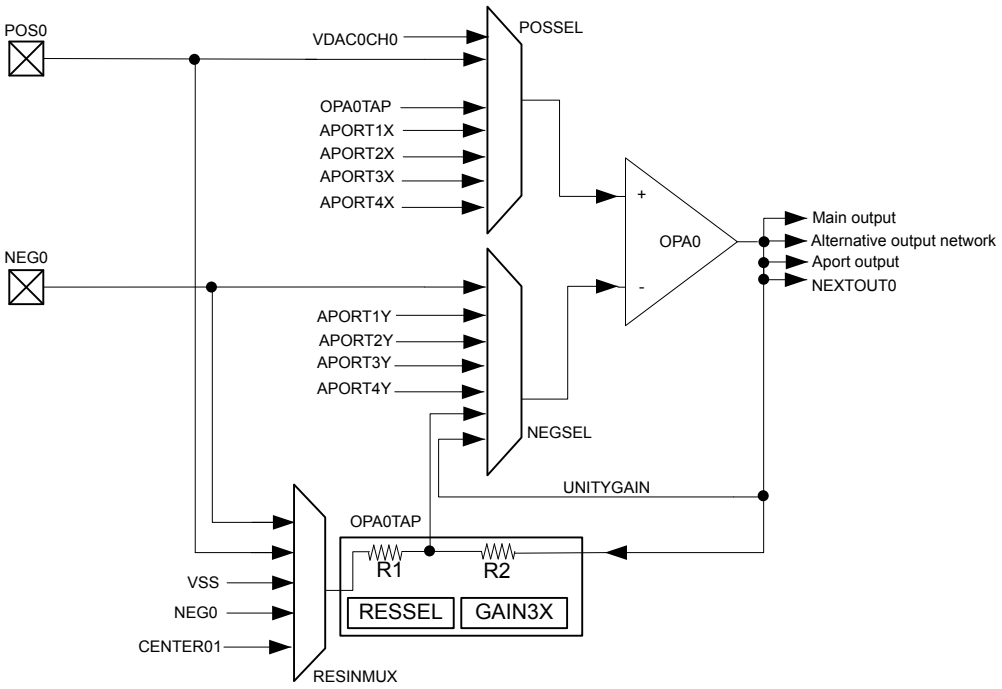
AN0038.1: Operational Amplifier (OPAMP)

This application note describes the theory of operational amplifiers (opamps) in general, and explains how to use the MCU Series 1, and Wireless SoC Series 1 operational amplifiers.

Useful code examples are included, where the theory behind is explained.

KEY POINTS

- Highly configurable general purpose opamps.
- Cascade connections between two, three, or four opamps.
- Opamp as VDAC output stage.
- Opamp as ADC or ACMP input front-end.
- Software examples.



1. Device Compatibility

This application note supports multiple device families, and some functionality is different depending on the device.

MCU Series 1 consists of:

- EFM32 Jade Gecko (EFM32JG12)
- EFM32 Pearl Gecko (EFM32PG12)
- EFM32 Giant Gecko (EFM32GG11/EFM32GG12)
- EFM32 Tiny Gecko (EFM32TG11)

Wireless SoC Series 1 consists of:

- EFR32 Blue Gecko (EFR32BG12/EFR32BG13/EFR32BG14)
- EFR32 Flex Gecko (EFR32FG12/EFR32FG13/EFR32FG14)
- EFR32 Mighty Gecko (EFR32MG12/EFR32MG13/EFR32MG14)

2. Operational Amplifier

2.1 Overview

An opamp is in its simplest form an amplifier with a theoretical infinite gain. The opamp has three main terminals (see Figure 2.1 A simple overview of an opamp on page 3) which are the positive (non-inverting) and negative (inverting) inputs in addition to the output. The general formula for an opamp is given by Figure 2.1 A simple overview of an opamp on page 3, where A_{OL} is the open loop gain which is infinite in an ideal opamp.

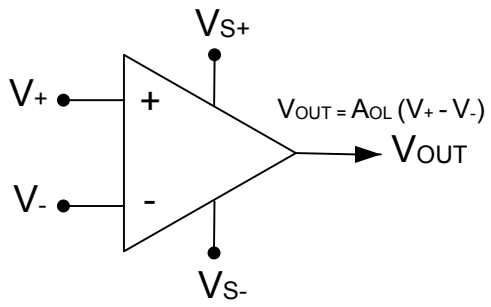


Figure 2.1. A simple overview of an opamp

In addition to the three main terminals, the opamp has two power supply pins V_{S+} and V_{S-}. These two voltage levels define the upper and lower voltage limit for the opamp output. Opamps also have a defined input range, which does not necessarily equal the output range. A rail-to-rail opamp is able to operate in the full range between V_{S+} and V_{S-}.

The versatility of the opamp lies in the various possible configurations that can be created by adding different feedback functions between the output and the positive and/or negative inputs. In the simplest form these feedbacks can consist of a few passive components like resistors or capacitors, but can also include more complex circuitry. These feedback connections allow many different kinds of filtering and amplification characteristics.

2.2 Characteristics

Open loop gain

The open loop gain (A_{OL}) in Figure 2.1 A simple overview of an opamp on page 3 is the gain without positive or negative feedback. Typical values range from 20,000 to 200,000 in physical devices.

Gain bandwidth product

In a real life opamp the gain is often a tradeoff against the bandwidth the opamp will be able to handle. Because of this relation, these two factors are often presented together as the gain bandwidth product.

Input offset voltage

The output V_{OUT} when $V_- = V_+$. This value is ideally 0, but in real life this value is nonzero.

Slew rate

The slew rate is the maximum change of voltage the opamp can output per time unit. Normally this is very high, which means that the output voltage can change rapidly. The slew rate is typically a few V/µs.

2.3 Ideal Operational Amplifier

An ideal opamp is a simplified model of a physical opamp. An ideal opamp is usually considered to have the following characteristics, and they are considered to hold for all input voltages:

- Infinite open loop gain
- Infinite input impedance
- Infinite slew rate
- Infinite bandwidth
- Zero output offset
- Zero input offset voltage
- Zero output impedance
- Zero noise

These characteristics are equivalent to the following two "golden" rules for ideal opamps:

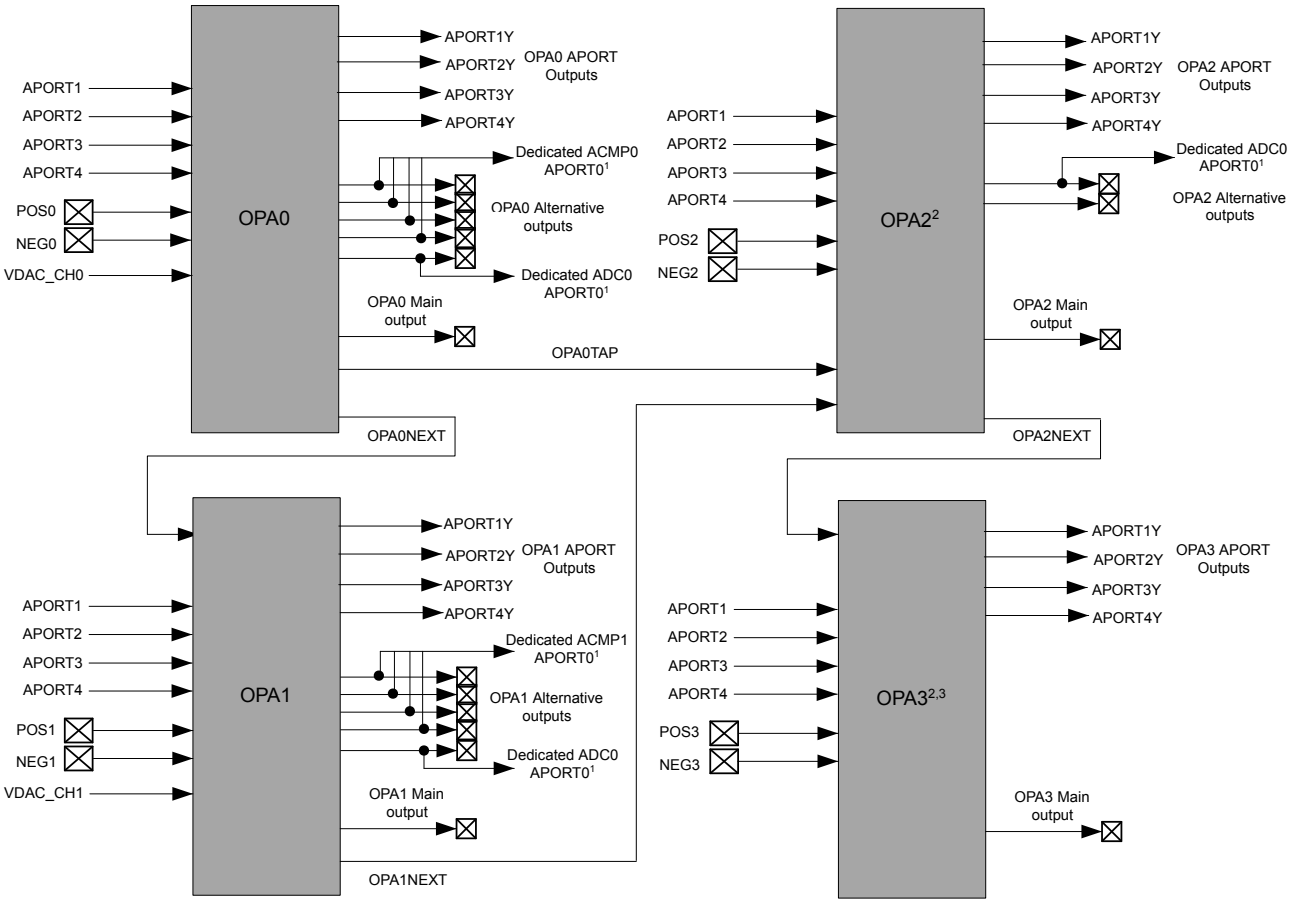
- The opamp will output whatever makes the two inputs as equal as possible ($V_+ = V_-$)
- The inputs draw no current ($I_+ = I_- = 0$)

Ideal opamps are used when calculating quantities for physical opamps. Since physical opamps have characteristics that are close to those of an ideal opamp, this is a good approximation in most cases.

3. Series 1 Operational Amplifier

3.1 Overview

The Series 1 (MCU Series 1, and Wireless SoC Series 1) device has up to four built in general purpose opamps, named OPA0, OPA1, OPA2, and OPA3 in Figure 3.1 A System Overview of the Series 1 Operational Amplifier on page 5. It is possible to connect these opamps together to make more complex configurations in addition to setting up either external or internal feedbacks.



[1]The dedicated ACMP0/ACMP1/ADC0 APORT0 are only for EFM32GG11 and EFM32TG11.
[2]There is no OPA2 and OPA3 on EFR32xG14 devices.
[3]There is no OPA3 on EFM32JG/PG12, EFR32xG12, and EFR32xG13 devices.

Figure 3.1. A System Overview of the Series 1 Operational Amplifier

All the Series 1 opamps are rail-to-rail for input and output. The positive source V_{S+} is equal to AVDD, and the negative source V_{S-} is connected to ground (see Figure 2.1 A simple overview of an opamp on page 3). A list containing electrical characteristics of the Series 1 opamps can be found in the device specific datasheets.

Note: Since two of the opamps (OPA0, OPA1) are part of the VDAC, the opamp configuration registers are located in the VDAC.

3.2 Input Configuration

The inputs to the opamps are controlled through a set of input muxes as shown in Figure 3.2 A Mux Overview of the Series 1 Operational Amplifier on page 6. The feedback path for each of the opamps includes an internal resistor ladder. This ladder can be configured to set the preferred gain value. It is also possible to bypass the ladder in unity gain mode.

The channel can be selected by configuring bitfields in VDACn_OPAx_MUX register as shown in Table 3.1 OPAMP Input Channel Selection on page 6.

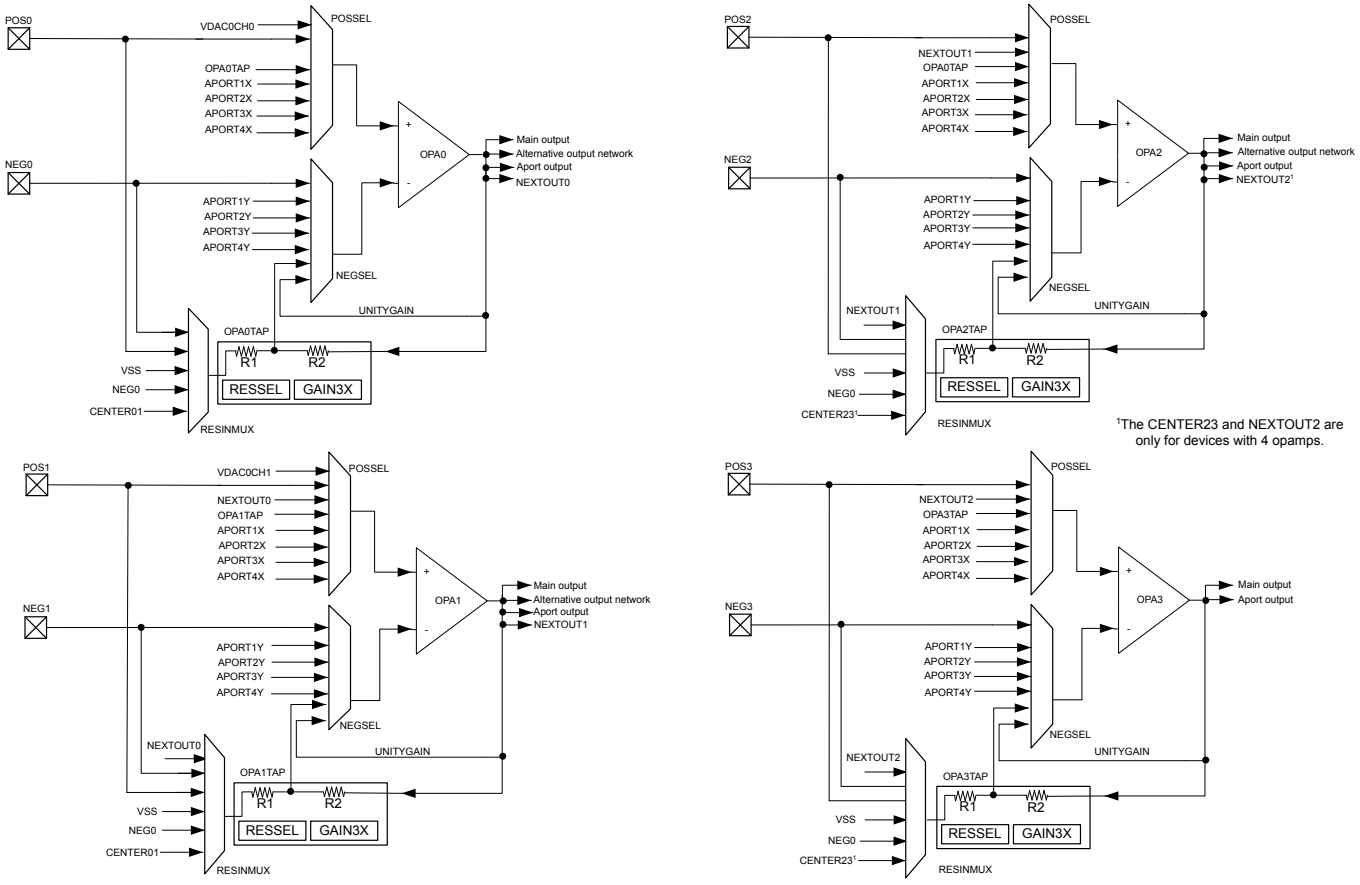


Figure 3.2. A Mux Overview of the Series 1 Operational Amplifier

Table 3.1. OPAMP Input Channel Selection

Bitfield	Selection	MUX connection
POSSEL	DISABLE	Input disabled
	DAC[1]	VDAC as input
	POSPAD[2]	POS pad as input
	OPANEXT[3]	NEXTOUT (x-1) as input
	OPATAP[4]	OPAxTAP as input
	APORT[1-4]XCHn	APORT as input

Bitfield	Selection	MUX connection
NEGSEL	DISABLE	Input disabled
	UG	Unity gain feedback path
	OPATAP	OPAxTAP as input
	NEGPAD	NEG pad as input
	APORT[1-4]YCHn	APORT as input
RESINMUX	DISABLE	Input disabled
	OPANEXT[3]	NEXTOUT (x-1) as input
	NEGPAD	NEG pad as input
	POSPAD[2]	POS pad as input
	COMPAD[5]	NEG pad of OPA0 as input
	CENTER[6]	Connected to resistor ladder of neighbor opamp
	VSS	VSS connected

Note:

1. The DAC option is only available on OPA0 and OPA1 (see 3.10 Opamp as VDAC Output Stage).
2. The POSPAD is the only direct input path for each OPA. When it is selected by RESINMUX, POSSEL should select other indirect path as its input.
3. It is not applicable on OPA0.
4. For OPA2, it is OPA0TAP for three opamp differential amplifier (see 4.11 Three Opamp Differential Amplifier).
5. Direct input to support common reference (see 3.9 Common Reference).
6. It is for fully differential instrumentation amplifier.

The pin-out of POS and NEG pads are listed in Table 3.3 Pin-out of OPAMP Input and Output on page 9 and APORT input channels (APORT[1-4]XCHn and APORT[1-4]YCHn) can be found in Analog Port (APORT) Client Maps section in the device specific data-sheets.

Note: The pins configured as OPAMP inputs should disable over voltage tolerance (OVT) feature by setting the corresponding bit in GPIO_Px_OVTDIS register to reduce any potential distortion introduced by the OVT circuitry.

3.3 Output Configuration

Each opamp has three outputs: the main output, an alternative output network with lower drive strength, and an APORT output with low drive strength. These three outputs can be configured as shown in Figure 3.3 An Output Stage Overview of the Series 1 Operational Amplifier on page 8.

The main and alternative outputs can be enabled or disabled by bitfields in VDACn_OPAx_OUT register as shown in Table 3.2 OPAMP Main and Alternative Output on page 8. The APORT output can drive the APORT selection mux by setting APORTOUTEN in VDACn_OPAx_OUT register. The APORT channel can be selected by configuring APORTOUTSEL in VDACn_OPAx_OUT register.

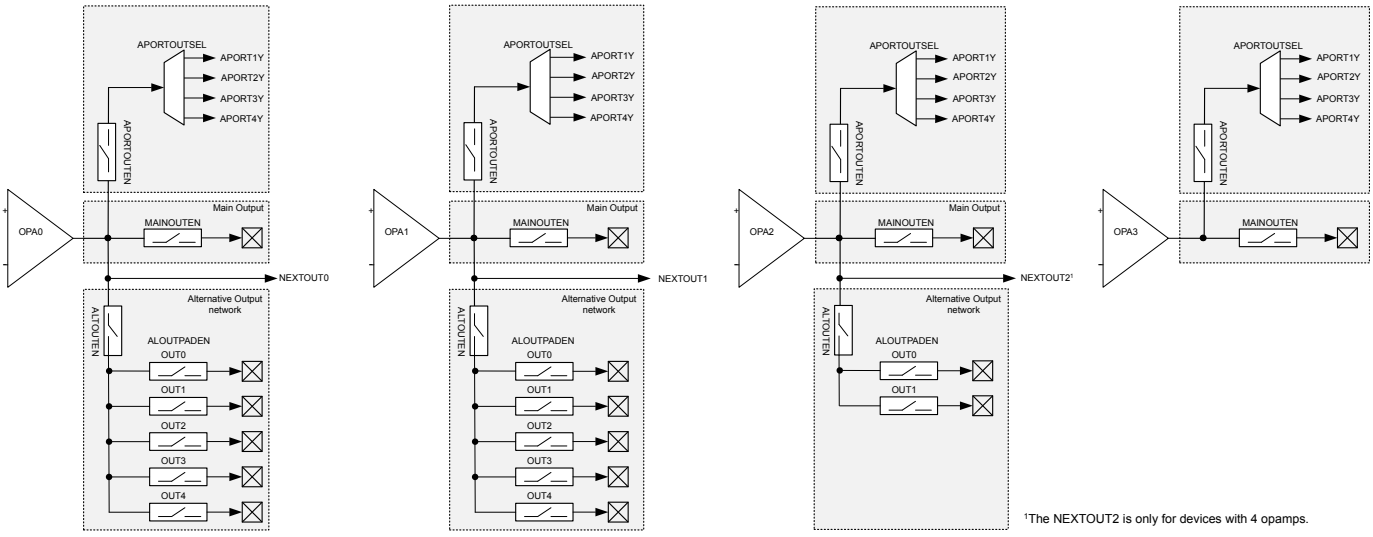


Figure 3.3. An Output Stage Overview of the Series 1 Operational Amplifier

Note: The OPAMPs can drive outputs above IOVDD and therefore the involved pads typically require over voltage tolerance (OVT) feature to be enabled.

Table 3.2. OPAMP Main and Alternative Output

MAINOUTEN	ALTOUTEN	SHORT[1]	Main Output	Alternative Output[2]
0	0	0	Disable	Disable
0	0	1	Disable	Disable
0	1	0	Disable	Enable
0	1	1	Enable	Enable
1	0	0	Enable	Disable
1	0	1	Enable	Enable
1	1	0	Enable	Enable
1	1	1	Enable	Enable

1. The main and alternate outputs of each opamp can be shorted together by setting the SHORT bitfield in VDACn_OPAx_OUT register.
2. The alternative output network consists of connections to pins which can be individually enabled by configuring ALTOUTPADEN bitfield in VDACn_OPAx_OUT register (see Figure 3.3 An Output Stage Overview of the Series 1 Operational Amplifier on page 8).

The pin-out of main and alternative outputs are listed in Table 3.3 Pin-out of OPAMP Input and Output on page 9 and APORT output channels (APORT[1-4]YCHn) can be found in Analog Port (APORT) Client Maps section in the device specific datasheets.

Table 3.3. Pin-out of OPAMP Input and Output

OPAMP Input and Output	EFM32GG11/12 and EFM32TG11[1]	EFM32JG/PG12 and EFR32xG12/13[1]	EFR32xG14[1]
OPA0_P	PC4	PA2	PA2
OPA0_N	PC5	PA4	PA4
OPA0_OUT	PB11	PA3	PA3
OPA0_OUTALT	• 0: PC0 • 1: PC1 • 2: PC2 • 3: PC3 • 4: PD0	• 0: PA5 • 1: PD13 • 2: PD15	• 0: PA5 • 1: PD13 • 2: PD15
OPA1_P	PD6	PD13	PD13
OPA1_N	PD7	PD15	PD15
OPA1_OUT	PB12	PD14	PD14
OPA1_OUTALT	• 0: PC12 • 1: PC13 • 2: PC14 • 3: PC15 • 4: PD1	• 0: PD12 • 1: PA2 • 2: PA4	• 0: PD12 • 1: PA2 • 2: PA4
OPA2_P	PD4	PB11	—
OPA2_N	PD3	PB13	—
OPA2_OUT	PD5	PB12	—
OPA2_OUTALT	• 0: PD0	• 0: PB9 • 1: PB10	—
OPA3_P	PC6	—	—
OPA3_N	PC7	—	—
OPA3_OUT	PD1	—	—

Note:
1. The available GPIOs for each device might vary, such differences are covered in the device specific datasheets.

3.4 APORT Request and Conflict Status

The request status of APORT buses is visible through the VDACn_OPAx_APORTREQ register. If an APORT bus conflict occurs, it is reported in the VDACn_OPAx_APORTCONFLICT register. An APORT conflict occurs if an opamp requests the same bus at the same time as another analog peripheral. In addition an APORT conflict is reported if any two of NEGSEL, POSSEL (in VDACn_OPAx_MUX register), or APORTOUTSEL (in VDACn_OPAx_OUT register) are configured to request the same APORT bus.

3.5 Gain Programming with Internal Resistor Ladder

The feedback path of each mux includes an internal resistor ladder (R2/R1) that can be used to select a set of gain values (see Figure 3.2 A Mux Overview of the Series 1 Operational Amplifier on page 6). Gain is configured by bitfields in VDACn_OPAx_MUX register as shown in Table 3.4 Internal Resistor Ladder for Gain Programming on page 10.

Note: The gain accuracy of the internal resistor feedback network is limited by the process variation of mux ON resistance and poly resistor which is used to implement R1 and R2.

Table 3.4. Internal Resistor Ladder for Gain Programming

RESSEL	GAIN3X[1]	R2/R1	Non-Inverting Gain (1 + R2/R1)	Inverting Gain (-R2/R1)
—	1	2	3.000	-2.000
0	0	1/3	1.333	-0.333
1	0	1	2.000	-1.000
2	0	5/3	2.667	-1.667
3	0	11/5	3.200	-2.200
4	0	3	4.000	-3.000
5	0	13/3	5.333	-4.333
6	0	7	8.000	-7.000
7	0	15	16.000	-15.000

Note:
1. When OPA0/1 is used as VDAC CH0/1 output stage, the GAIN3X is set to 1 for VDAC to work properly.

3.6 Opamp Characteristics

The opamp characteristics are programmable with bitfields in VDACn_OPAx_CTRL register as shown in Table 3.5 OPAMP Characteristics on page 10.

Table 3.5. OPAMP Characteristics

Bitfield	Usage
DRIVESTRENGTH	To program drive strength. • 0: Lower accuracy with Low drive strength • 1: Low accuracy with Low drive strength • 2: High accuracy with High drive strength • 3: Higher accuracy with High drive strength
INCBW[1]	Set to scale the unity gain bandwidth by factor of 2.5. It makes opamp operate faster for closed-loop gain setting greater than 3x but the opamp is not unity gain stable.
HCMDIS[1]	Set to disable rail-to-rail on input (High Common Mode), while output still remains rail-to-rail. The opamp input voltage is restricted between VSS and VDD - 1.2V to improve linearity of the opamp.
OUTSCALE	To scale opamp output driving strength. • 0: Full output driving strength • 1: Half output driving strength

Note:
1. When OPA0/1 is used as VDAC CH0/1 output stage, the INCBW and HCMDIS are set to 1.

3.7 Timing

The warmup time, settle time, and startup delay of opamp are programmable with bitfields in VDACn_OPAx_TIMER register as shown in Table 3.6 OPAMP Timing on page 11.

Table 3.6. OPAMP Timing

Bitfield	DRIVES-TRENGTH	Time (µs)	Remark
WARMUPTIME	0	100	• The warm up period depends on the selected DRIVESTRENGTH in VDACn_OPAx_CTRL register. • The OPAxWARM bit in VDACn_STATUS register is set when the warmup period has completed. • Up to 127µs.
	1	85	
	2	8	
	3	6	
SETTLETIME[1]	0	60	• The settling period depends on the load at opamp output and selected DRIVESTRENGTH in VDACn_OPAx_CTRL register. • The OPAxOUTVALID bit in VDACn_STATUS register is set when the settle period has completed. • Up to 1023µs.
	1	25	
	2	3	
	3	1	
STARTDLY	—	0 - 63µs	• Used only in PRS mode. • The opamp is warmed up after STARTDLY + WARMUPTIME. • The output is settled after STARTDLY + WARMUPTIME + SETTLETIME.

Note:
1. The SETTLETIME specifies settings for a load of 1KOhm and 75pF.

3.8 Enable Source and PRS Output

The opamp can be enabled by software or PRS, which is configured by bitfields in VDACn_OPAx_CTRL register as shown in Table 3.7 OPAMP Enable Source on page 12.

Table 3.7. OPAMP Enable Source

PRSEN	PRSMODE	PRSSEL[1]	Remark
0	—	—	The OPAx is enabled by OPAxEN in VDACn_CMD register.
1	0 (PULSED)	PRSCHn	The OPAx is enabled by PRS input channel selected by PRSSEL. The opamp is turned on by the positive edge of PRS pulse and stays on based on the timing configurations in VDACn_OPAx_TIMER register.
1	1 (TIMED)	PRSCHn	The OPAx is enabled by PRS input channel selected by PRSSEL. The opamp is turned on by the positive edge of PRS pulse and stays on until PRS goes low. The PRS pulse should be long enough for opamp to warmup and settle.[2]

Note:
1. The available PRS channels (0 to n) can be found in device reference manual.
2. The OPAxPRSTIMEDERR interrupt flag in VDACn_IF register indicates a protocol error when the opamp is triggered in PRS TIMED mode. This flag is set if the negative edge of the PRS pulse came before the opamp output was valid.

One of the two asynchronous PRS outputs can be enabled for each opamp, which is configured by PRSOUTMODE in VDACn_OPAx_CTRL register as shown in Table 3.8 OPAMP PRS Output on page 12.

Table 3.8. OPAMP PRS Output

Enable Source	PRSOUTMODE	PRS Output
Software	0 (WARM)	The PRS output goes high when opamp is warm and output is enabled.
	1 (OUTVALID)	The PRS output goes high when opamp output is settle and valid.
PRS PULSED	0 (WARM)	• The PRS output goes high when opamp is warm and output is enabled (STARTDLY + WARMUPTIME). • The PRS output goes low when opamp output is invalid.
	1 (OUTVALID)	• The PRS output goes high when opamp output is settle and valid (STARTDLY + WARMUP-TIME + SETTLETIME). • The PRS output goes low when opamp output is invalid.
PRS TIMED	0 (WARM)	• The PRS output goes high when opamp is warm and output is enabled (STARTDLY + WARMUPTIME). • The PRS output goes low when PRS pulse goes low.
	1 (OUTVALID)	• The PRS output goes high when opamp output is settle and valid (STARTDLY + WARMUP-TIME + SETTLETIME). • The PRS output goes low when PRS pulse goes low.

3.9 Common Reference

It is possible to configure all opamps to have a common reference by setting the RESINMUX bitfield to COMPAD in VDACn_OPAx_MUX register. When RESINMUX of all opamps is set to COMPAD mode, the NEGPAD input of OPA0 is used.

3.10 Opamp as VDAC Output Stage

The OPA0 and OPA1 are used as VDAC channel 0 (CH0) and channel 1 (CH1) output stage as shown in Figure 3.4 VDAC Output Stage on page 13, when the corresponding channel is enabled. In that configuration, user should not use OPA0EN/OPA1EN or OPA0DIS/OPA1DIS in VDACn_CMD register to enable or disable the OPA0 or OPA1, because they are under the control of CH0EN/CH1EN and CH0DIS/CHI1DIS in VDACn_CMD register. If both VDAC channels are used, OPA0 and OPA1 cannot be used as stand-alone opamp.

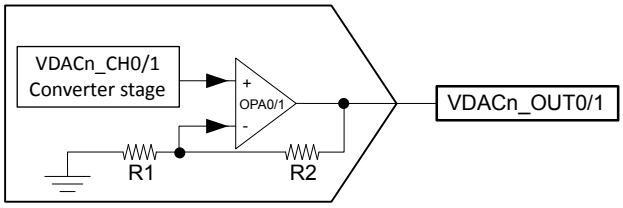


Figure 3.4. VDAC Output Stage

4. Software Examples

4.1 Overview

This application note includes software examples demonstrating how to use the opamp in different configurations. Each of these configurations have a corresponding C software example written for the EFM32PG12, EFM32GG11, EFM32TG11 Starter Kits and EFR32MG12, EFR32MG13, EFR32FG14 radio boards.

The resistors in the examples are internal, so no external resistors or other external components are needed. Table on each example gives an overview of all inputs and outputs (defined in `opamp_config.h`) needed to set up the configurations. In all calculations the opamps are assumed to be ideal.

The examples use the APORT to route signals to the opamp's input nodes and to route the opamp's output to a GPIO pin. To change the pin mappings for the examples, refer to the Analog Port (APORT) Client Maps section of your device's datasheet and change the mappings accordingly by using `EMLIB` type defines (e.g., `opaPosSelAPORT3XCH10`).

The board connections (EXP header, J101, J012, and WSTK breakout pin) for the opamps can be found in the schematic and assembly drawing of the associated Starter Kit (STK) or Wireless Starter Kit (WSTK) main board.

The board controller on the STK or WSTK provides a virtual COM port (CDC) interface when connected to a computer. The on board device can connect to this serial port interface and communicate directly (baud rate 115200-8-N-1) with the host computer terminal program (e.g., Tera Term).

The examples in following sections are selected by the main menu in the host computer terminal program as below. There is only two opamps in EFR32xG14 devices so `Three OPA differential amplifier` example is not available on EFR32FG14 radio board.

```
OPAMP Examples
Press a for General OPA mode
Press b for Unity gain voltage follower
Press c for Non-inverting amplifier
Press d for Inverting amplifier
Press e for Cascaded non-inverting amplifier
Press f for Cascaded inverting amplifier
Press g for Two OPA differential amplifier
Press h for OPA as ACMP input front-end
Press i for OPA as ADC input front-end
Press j for Three OPA differential amplifier
```

The on board device will enter Energy Mode 2 (EM2) or Energy Mode 3 (EM3) when the selected example is running. The user needs to reset the STK or WSTK to select another example.

```
Unity gain voltage follower at EM3
Reset the STK/WSTK to select another example
```

To measure the current consumption in EM2 or EM3, the debugger must be disconnected from the IDE. Then, switch the power selector on the STK or WSTK to the "BAT" position and then back to the "AEM" position to provide a Power-on Reset (POR) to the DC-DC converter.

4.2 General Opamp Mode

The simplest configuration without internal feedback is the general opamp mode. All pins are available externally for custom configuration.

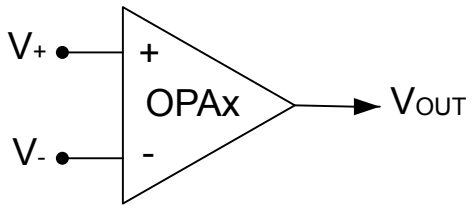


Figure 4.1. General Opamp Mode

The `opaGeneralMode()` function in `opamp.c` is used to configure the general opamp mode.

```
void opaGeneralMode(OPAMP_TypeDef opaSelect,            // OPAx - OPA0/1/2/3
                    OPAMP_PosSel_TypeDef posInput,      // V+
                    OPAMP_NegSel_TypeDef negInput,      // V-
                    OPAMP_OutMode_TypeDef opaOutput,    // VOUT
                    uint32_t altOutMask)                // Enable bit mask if VOUT is alternate o/p
```

Example of `opaGeneralMode()` usage on SLSTK3701A:

```
opaGeneralMode(OPA1, opaPosSelPosPad, opaNegSelNegPad, opaOutModeAlt, 0x0100);
```

Press "a" to run this example and the device stays in EM3.

```
General OPA mode at EM3
Reset the STK/WSTK to select another example
```

Table 4.1. Starter Kit/Radio Board for Inverting Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A (OPA1)	EFM32TG11B520F128GM80	V_+	PD6 — OPA1_P	Expansion header pin 16
		V_-	PD7 — OPA1_N	Expansion header pin 15
		V_{OUT}	PC14 — OPA1_OUTALT #2	Expansion header pin 12
SLSTK3701A[1] (OPA1)	EFM32GG11B820F2048GL192	V_+	PD6 — OPA1_P	J102 pin 26
		V_-	PD7 — OPA1_N	J102 pin 28
		V_{OUT}	PD1 — OPA1_OUTALT #4	J102 pin 22
SLSTK3402A (OPA0)	EFM32PG12B500F1024GL125	V_+	PC9 — APORT2XCH9	Expansion header pin 3
		V_-	PC10 — APORT2YCH10	Expansion header pin 16
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A (OPA0)	EFR32MG12P432F1024GL125	V_+	PC9 — APORT2XCH9	Expansion header pin 13
		V_-	PC10 — APORT2YCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
• BRD4159A (OPA0) • BRD4257A (OPA0)	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_+	PC9 — APORT2XCH9	Expansion header pin 10
		V_-	PC10 — APORT2YCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

Note:

1. On the back side of the SLSTK3402A Starter Kit, there is a footprint of OPA1 to connect external components.

4.3 Unity Gain Voltage Follower

The simplest configuration with feedback is the voltage follower configuration. Here the output is routed directly back to the inverting input. This configuration outputs the same voltage value as the input, V_{IN}. The voltage follower is commonly used as a buffer to increase the drive strength to drive higher loads.

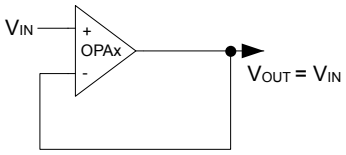


Figure 4.2. Unity Gain Voltage Follower

The `opaUnityGain()` function in `opamp.c` is used to configure the unity gain voltage follower.

```
void opaUnityGain(OPAMP_TypeDef opaSelect,              // OPAx - OPA0/1/2/3
                  OPAMP_PosSel_TypeDef posInput,        // VIN
                  OPAMP_OutMode_TypeDef opaOutput,      // VOUT
                  uint32_t altOutMask)                  // Enable bit mask if VOUT is alternate o/p
```

Example of `opaUnityGain()` usage on SLSTK3701A:

```
opaUnityGain(OPA0, opaPosSelAPORT3XCH10, opaOutModeAPORT4YCH12, 0);
```

Press "b" to run this example and the device stays in EM3.

```
Unity gain voltage follower at EM3
Reset the STK/WSTK to select another example
```

Table 4.2. Starter Kit/Radio Board for Unity Gain Voltage Follower Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A (OPA1)	EFM32TG11B520F128GM80	V_{IN}	PD6 — OPA1_P	Expansion header pin 16
		V_{OUT}	PC14 — OPA1_OUTALT #2	Expansion header pin 12
SLSTK3701A (OPA0)	EFM32GG11B820F2048GL192	V_{IN}	PE10 — APORT3XCH10	Expansion header pin 4
		V_{OUT}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A (OPA0)	EFM32PG12B500F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 3
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A (OPA0)	EFR32MG12P432F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 13
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A (OPA0) • BRD4257A (OPA0)	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 10
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

4.4 Non-inverting Amplifier

This configuration amplifies the input signal V_{IN} and the output signal V_{OUT} is determined by R1 and R2 as shown in Figure 4.3 Non-inverting Amplifier on page 18. The V_{IN} is connected to the non-inverting input and R2/R1 = 1 in this example.

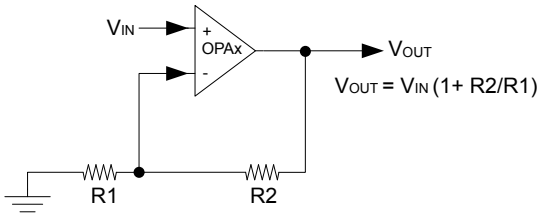


Figure 4.3. Non-inverting Amplifier

The `opaNonInvertAmp()` function in `opamp.c` is used to configure the non-inverting amplifier.

```
void opaNonInvertAmp(OPAMP_TypeDef opaSelect,                    // OPAx - OPA0/1/2/3
                     OPAMP_PosSel_TypeDef posInput,              // VIN
                     OPAMP_ResSel_TypeDef resLadderRatio,        // R2/R1
                     OPAMP_OutMode_TypeDef opaOutput,            // VOUT
                     uint32_t altOutMask)                        // Enable bit mask if VOUT is alternate o/p
```

Example of `opaNonInvertAmp()` usage on SLSTK3701A:

```
opaNonInvertAmp(OPA0, opaPosSelAPORT3XCH10, opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

Press "c" to run this example and the device stays in EM3.

```
Non-inverting amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.3. Starter Kit/Radio Board for Non-inverting Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A (OPA1)	EFM32TG11B520F128GM80	V_{IN}	PD6 — OPA1_P	Expansion header pin 16
		V_{OUT}	PC14 — OPA1_OUTALT #2	Expansion header pin 12
SLSTK3701A (OPA0)	EFM32GG11B820F2048GL192	V_{IN}	PE10 — APORT3XCH10	Expansion header pin 4
		V_{OUT}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A (OPA0)	EFM32PG12B500F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 3
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A (OPA0)	EFR32MG12P432F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 13
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A (OPA0) • BRD4257A (OPA0)	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 10
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

4.5 Inverting Amplifier

This configuration inverts the input signal V_{IN} around the POS signal applied to the positive input as shown in Figure 4.4 Inverting Amplifier on page 19. The ratio R2/R1 determines the inverting gain and unity gain is achieved when R2/R1 = 1. This example uses R2/R1 = 1 and V_{OUT} = 2POS - V_{IN}.

The value of POS usually equals AVDD/2 on MCU Series 1, and Wireless SoC Series 1 devices, giving full range on the output.

This is exactly the same configuration as the non-inverting amplifier, except that V_{IN} is connected to the inverting input, and the non-inverting input is connected to the POS reference.

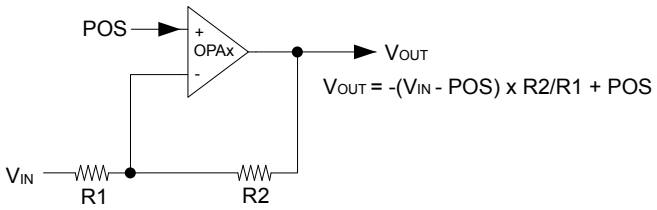


Figure 4.4. Inverting Amplifier

The `opaInvertAmp()` function in `opamp.c` is used to configure the inverting amplifier.

```
void opaInvertAmp(OPAMP_TypeDef opaSelect,                    // OPAx - OPA0/1/2/3
                  OPAMP_ResInMux_TypeDef resInInput,          // VIN
                  OPAMP_PosSel_TypeDef posInput,              // POS
                  OPAMP_ResSel_TypeDef resLadderRatio,        // R2/R1
                  OPAMP_OutMode_TypeDef opaOutput,            // VOUT
                  uint32_t altOutMask)                        // Enable bit mask if VOUT is alternate o/p
```

Example of `opaInvertAmp()` usage on SLSTK3701A:

```
opaInvertAmp(OPA0, opaResInMuxNegPad, opaPosSelAPORT3XCH10, opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

Press "d" to run this example and the device stays in EM3.

```
Inverting amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.4. Starter Kit/Radio Board for Inverting Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A (OPA1)	EFM32TG11B520F128GM80	V_{IN}	PD7 — OPA1_N	Expansion header pin 15
		POS	PD6 — OPA1_P	Expansion header pin 16
		V_{OUT}	PC14 — OPA1_OUTALT #2	Expansion header pin 12
SLSTK3701A (OPA0)	EFM32GG11B820F2048GL192	V_{IN}	PC5 — OPA0_N	Expansion header pin 9
		POS	PE10 — APORT3XCH10	Expansion header pin 4
		V_{OUT}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A (OPA0)	EFM32PG12B500F1024GL125	V_{IN}	PA2 — OPA0_P	J101 pin 9
		POS	PC9 — APORT2XCH9	Expansion header pin 3
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
BRD4161A (OPA0)	EFR32MG12P432F1024GL125	V_{IN}	PA2 — OPA0_P	WSTK breakout pin P35
		POS	PC9 — APORT2XCH9	Expansion header pin 13
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A (OPA0) • BRD4257A (OPA0)	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PA2 — OPA0_P	Expansion header pin 3
		POS	PC9 — APORT2XCH9	Expansion header pin 10
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

4.6 Cascaded Non-inverting Amplifier

This configuration chains up to four non-inverting amplifiers to create a higher overall gain. The output of OPA0 is routed to the non-inverting input of OPA1. Similarly, the outputs of OPA1 and OPA2 are routed to the non-inverting inputs of OPA2 and OPA3. If one opamp can amplify the signal by a factor of F, then this cascaded configuration can amplify by a factor of F^4. This example uses two stages cascaded non-inverting amplifier with R2/R1 = 1 for all resistor ladders (F = 2).

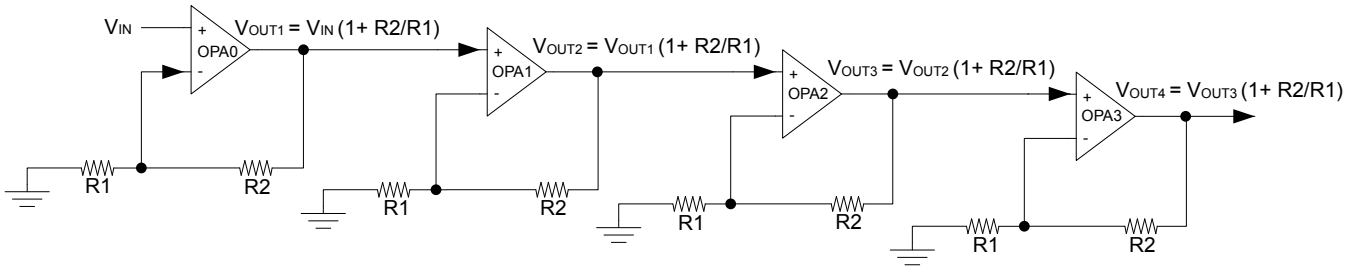


Figure 4.5. Cascaded Non-inverting Amplifier

The `opaCascadeTwoNonInvertAmp()` function in `opamp.c` is used to configure the two stages cascaded non-inverting amplifier.

```
void opaCascadeTwoNonInvertAmp(OPAMP_TypeDef firstStage,        // OPA0/1/2
                               OPAMP_PosSel_TypeDef posInput0,  // VIN
                               OPAMP_ResSel_TypeDef resSel0,    // OPA0/1/2 R2/R1
                               OPAMP_ResSel_TypeDef resSel1,    // OPA1/2/3 R2/R1
                               OPAMP_OutMode_TypeDef opaOutput, // VOUT2
                               uint32_t altOutMask)             // Enable bit mask if VOUT2 is alternate o/p
```

Example of `opaCascadeTwoNonInvertAmp()` usage on SLSTK3701A:

```
opaCascadeTwoNonInvertAmp(OPA0, opaPosSelAPORT3XCH10, opaResSelR2eqR1,
                          opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

The `opaCascadeThreeNonInvertAmp()` function in `opamp.c` is used to configure the three stages cascaded non-inverting amplifier.

```
void opaCascadeThreeNonInvertAmp(OPAMP_TypeDef firstStage,        // OPA0/1
                                 OPAMP_PosSel_TypeDef posInput0,  // VIN
                                 OPAMP_ResSel_TypeDef resSel0,    // OPA0/1 R2/R1
                                 OPAMP_ResSel_TypeDef resSel1,    // OPA1/2 R2/R1
                                 OPAMP_ResSel_TypeDef resSel2,    // OPA2/3 R2/R1
                                 OPAMP_OutMode_TypeDef opaOutput, // VOUT3
                                 uint32_t altOutMask)             // Enable bit mask if VOUT3 is alternate o/p
```

Example of `opaCascadeThreeNonInvertAmp()` usage:

```
opaCascadeThreeNonInvertAmp(OPA0, opaPosSelAPORT3XCH10, opaResSelR2eqR1, opaResSelR2eqR1,
                            opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

The `opaCascadeFourNonInvertAmp()` function in `opamp.c` is used to configure the four stages cascaded non-inverting amplifier.

```
void opaCascadeFourNonInvertAmp(OPAMP_PosSel_TypeDef posInput0,  // VIN
                                OPAMP_ResSel_TypeDef resSel0,    // OPA0 R2/R1
                                OPAMP_ResSel_TypeDef resSel1,    // OPA1 R2/R1
                                OPAMP_ResSel_TypeDef resSel2,    // OPA2 R2/R1
                                OPAMP_ResSel_TypeDef resSel3,    // OPA3 R2/R1
                                OPAMP_OutMode_TypeDef opaOutput, // VOUT4
                                uint32_t altOutMask)             // Enable bit mask if VOUT4 is alternate o/p
```

Example of `opaCascadeFourNonInvertAmp()` usage:

```
opaCascadeFourNonInvertAmp(opaPosSelAPORT3XCH10, opaResSelR2eqR1, opaResSelR2eqR1, opaResSelR2eqR1,
                           opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

Press "e" to run this example and the device stays in EM3.

```
Cascaded two non-inverting amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.5. Starter Kit/Radio Board for Cascaded Non-inverting Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A	EFM32TG11B520F128GM80	V_{IN}	PD6 — OPA1_P	Expansion header pin 16
		V_{OUT2}	PD5 — OPA2_OUT	Expansion header pin 11
SLSTK3701A	EFM32GG11B820F2048GL192	V_{IN}	PE10 — APORT3XCH10	Expansion header pin 4
		V_{OUT2}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A	EFM32PG12B500F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 3
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A	EFR32MG12P432F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 13
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A • BRD4257A	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 10
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 16

4.7 Cascaded Inverting Amplifier

This configuration consists of up to four inverting amplifiers put together to achieve higher accumulated gain. This example uses two stages cascaded inverting amplifier with R2/R1 = 1 for all resistor ladders.

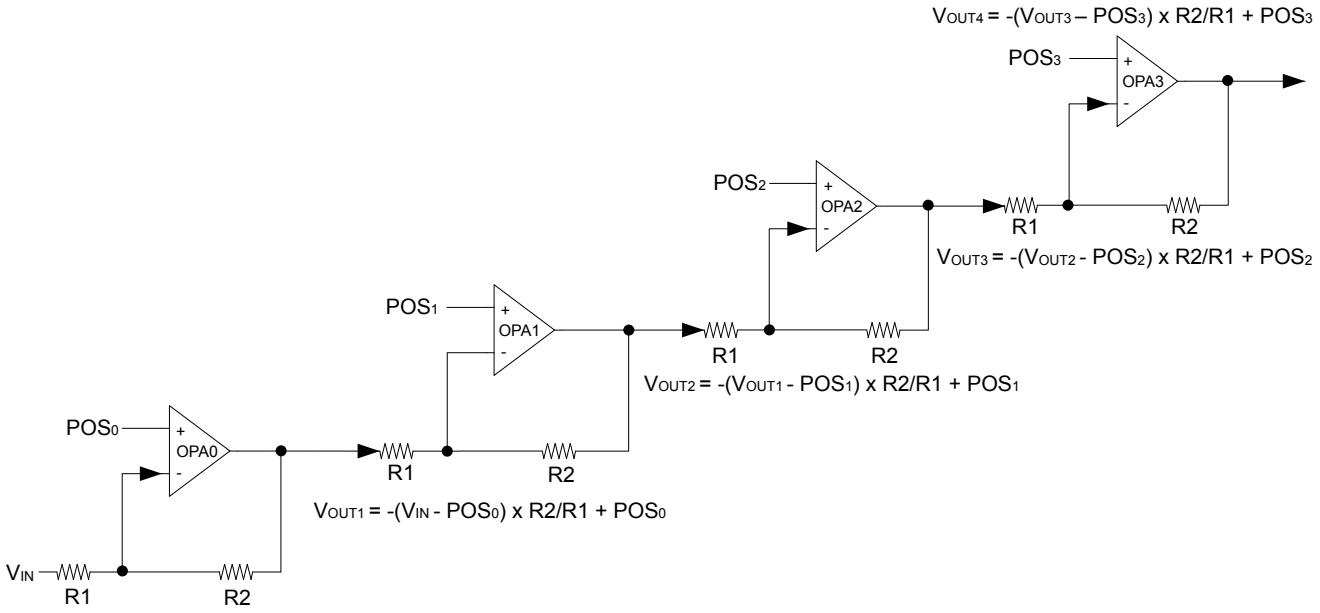


Figure 4.6. Cascaded Inverting Amplifier

The `opaCascadeTwoInvertAmp()` function in `opamp.c` is used to configure the two stages cascaded inverting amplifier.

```
void opaCascadeTwoInvertAmp(OPAMP_TypeDef firstStage,           // OPA0/1/2
                            OPAMP_ResInMux_TypeDef resInInput,  // VIN
                            OPAMP_PosSel_TypeDef posInput0,     // POS0
                            OPAMP_ResSel_TypeDef resSel0,       // OPA0/1/2 R2/R1
                            OPAMP_PosSel_TypeDef posInput1,     // POS1
                            OPAMP_ResSel_TypeDef resSel1,       // OPA1/2/3 R2/R1
                            OPAMP_OutMode_TypeDef opaOutput,    // VOUT2
                            uint32_t altOutMask)                // Enable bit mask if VOUT2 is alternate o/p
```

Example of `opaCascadeTwoInvertAmp()` usage on SLSTK3701A:

```
opaCascadeTwoInvertAmp(OPA0, opaResInMuxNegPad, opaPosSelAPORT3XCH10, opaResSelR2eqR1, opaPosSelAPORT4XCH11,
                       opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

The `opaCascadeThreeInvertAmp()` function in `opamp.c` is used to configure the three stages cascaded inverting amplifier.

```
void opaCascadeThreeInvertAmp(OPAMP_TypeDef firstStage,          // OPA0/1
                              OPAMP_ResInMux_TypeDef resInInput, // VIN
                              OPAMP_PosSel_TypeDef posInput0,    // POS0
                              OPAMP_ResSel_TypeDef resSel0,      // OPA0/1 R2/R1
                              OPAMP_PosSel_TypeDef posInput1,    // POS1
                              OPAMP_ResSel_TypeDef resSel1,      // OPA1/2 R2/R1
                              OPAMP_PosSel_TypeDef posInput2,    // POS2
                              OPAMP_ResSel_TypeDef resSel2,      // OPA2/3 R2/R1
                              OPAMP_OutMode_TypeDef opaOutput,   // VOUT3
                              uint32_t altOutMask)               // Enable bit mask if VOUT3 is alternate o/p
```

Example of `opaCascadeThreeInvertAmp()` usage:

```
opaCascadeThreeInvertAmp(OPA0, opaResInMuxNegPad, opaPosSelAPORT3XCH8, opaResSelR2eqR1, opaPosSelAPORT4XCH9,
                         opaResSelR2eqR1, opaPosSelAPORT1XCH12, opaResSelR2eqR1, opaOutModeAPORT1YCH13, 0);
```

The `opaCascadeFourInvertAmp()` function in `opamp.c` is used to configure the four stages cascaded inverting amplifier.

```
void opaCascadeFourInvertAmp(OPAMP_ResInMux_TypeDef resInInput, // VIN
                             OPAMP_PosSel_TypeDef posInput0,    // POS0
                             OPAMP_ResSel_TypeDef resSel0,      // OPA0 R2/R1
                             OPAMP_PosSel_TypeDef posInput1,    // POS1
                             OPAMP_ResSel_TypeDef resSel1,      // OPA1 R2/R1
                             OPAMP_PosSel_TypeDef posInput2,    // POS2
                             OPAMP_ResSel_TypeDef resSel2,      // OPA2 R2/R1
                             OPAMP_PosSel_TypeDef posInput3,    // POS3
                             OPAMP_ResSel_TypeDef resSel3,      // OPA3 R2/R1
                             OPAMP_OutMode_TypeDef opaOutput,   // VOUT4
                             uint32_t altOutMask)               // Enable bit mask if VOUT4 is alternate o/p
```

Example of `opaCascadeFourInvertAmp()` usage:

```
opaCascadeFourInvertAmp(opaResInMuxNegPad, opaPosSelAPORT3XCH8, opaResSelR2eqR1, opaPosSelAPORT4XCH9,
                        opaResSelR2eqR1, opaPosSelAPORT1XCH12, opaResSelR2eqR1, opaPosSelAPORT2XCH13,
                        opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

Press "f" to run this example and the device stays in EM3.

```
Cascaded two inverting amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.6. Starter Kit/Radio Board for Cascaded Inverting Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A	EFM32TG11B520F128GM80	V_{IN}	PD7 — OPA1_N	Expansion header pin 15
		POS_0	PD6 — OPA1_P	Expansion header pin 16
		POS_1	PA14 — APORT1XCH14	J101 pin 5
		V_{OUT2}	PD5 — OPA2_OUT	Expansion header pin 11
SLSTK3701A	EFM32GG11B820F2048GL192	V_{IN}	PC5 — OPA0_N	Expansion header pin 9
		POS_0	PE10 — APORT3XCH10	Expansion header pin 4
		POS_1	PE11 — APORT4XCH11	Expansion header pin 6
		V_{OUT2}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A	EFM32PG12B500F1024GL125	V_{IN}	PA2 — OPA0_P	J101 pin 9
		POS_0	PC9 — APORT2XCH9	Expansion header pin 3
		POS_1	PC10 — APORT1XCH10	Expansion header pin 16
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A	EFR32MG12P432F1024GL125	V_{IN}	PA2 — OPA0_P	WSTK breakout pin P35
		POS_0	PC9 — APORT2XCH9	Expansion header pin 13
		POS_1	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A • BRD4257A	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PA2 — OPA0_P	Expansion header pin 3
		POS_0	PC9 — APORT2XCH9	Expansion header pin 10
		POS_1	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT2}	PC11 — APORT1YCH11	Expansion header pin 16

4.8 Two Opamp Differential Amplifier

This configuration uses two opamps to output the weighted difference between the two inputs as shown in Figure 4.7 Two Opamp Differential Amplifier on page 25. The result is given as a differential output, V_{DIFF}, between the two output pins V_{OUT} and V_2. The first opamp is just used as a voltage follower, and the feedback in the second opamp is enabled. This example uses R2/R1 = 1 so V_{DIFF} = (V_2 - V_1).

As the Figure 4.7 Two Opamp Differential Amplifier on page 25 shows, it is possible to use OPA0 and OPA1, or OPA1 and OPA2, or OPA2 and OPA3 to set up this configuration.

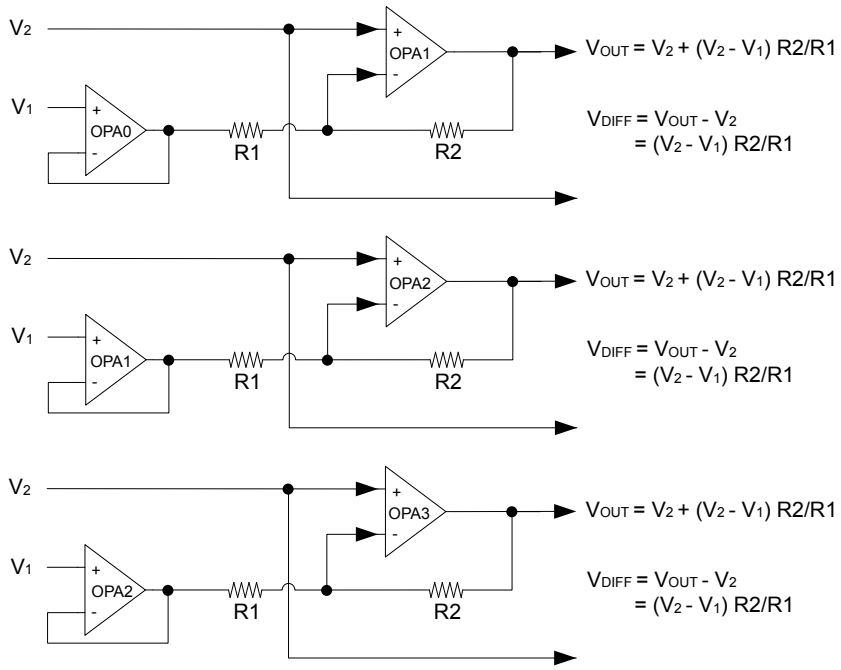


Figure 4.7. Two Opamp Differential Amplifier

The `opaTwoDiffAmp()` function in `opamp.c` is used to configure the two opamp differential amplifier.

```
void opaTwoDiffAmp(OPAMP_TypeDef firstStage,            // OPA0/1/2
                   OPAMP_PosSel_TypeDef posInput0,      // V1
                   OPAMP_PosSel_TypeDef posInput1,      // V2
                   OPAMP_ResSel_TypeDef resSel1,        // OPA1/2/3 R2/R1
                   OPAMP_OutMode_TypeDef opaOutput,     // VOUT
                   uint32_t altOutMask)                 // Enable bit mask if VOUT is alternate o/p
```

Example of `opaTwoDiffAmp()` usage on SLSTK3701A:

```
opaTwoDiffAmp(OPA0, opaPosSelAPORT3XCH10, opaPosSelAPORT4XCH11, opaResSelR2eqR1, opaOutModeAPORT4YCH12, 0);
```

Press "g" to run this example and the device stays in EM3.

```
Two OPA differential amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.7. Starter Kit/Radio Board for Two Opamp Differential Amplifier Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A	EFM32TG11B520F128GM80	V_1	PD6 — OPA1_P	Expansion header pin 16
		V_2	PA14 — APORT1XCH14	J101 pin 5
		V_{OUT}	PD5 — OPA2_OUT	Expansion header pin 11
SLSTK3701A	EFM32GG11B820F2048GL192	V_1	PE10 — APORT3XCH10	Expansion header pin 4
		V_2	PE11 — APORT4XCH11	Expansion header pin 6
		V_{OUT}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A	EFM32PG12B500F1024GL125	V_1	PC9 — APORT2XCH9	Expansion header pin 3
		V_2	PC10 — APORT1XCH10	Expansion header pin 16
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A	EFR32MG12P432F1024GL125	V_1	PC9 — APORT2XCH9	Expansion header pin 13
		V_2	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16
• BRD4159A • BRD4257A	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_1	PC9 — APORT2XCH9	Expansion header pin 10
		V_2	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

4.9 Opamp as ACMP Input Front-end

The OPA0 and OPA1 can route their outputs directly to ACMPn positive or negative input as shown in Figure 4.8 ACMP Input Front-end on page 27. The opamp output selection is done by POSSEL and NEGSEL bitfields in ACMPn_INPUTSEL register. The ACMPn can use opamps as its input front-end. When OPA0/1 is used as the VDACn output stage, the VDACn can also route its output VDACn_OUT0/1 to the ACMPn input. The OPA2 and OPA3 do not have such direct connection path.

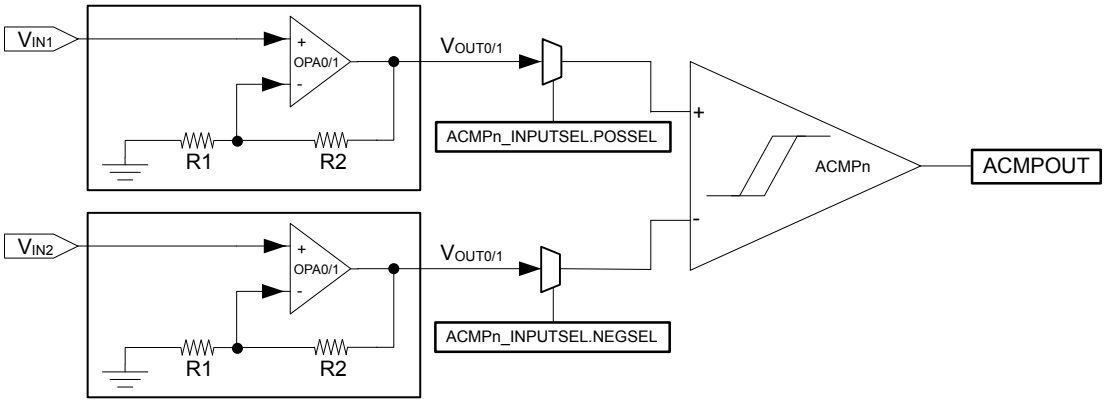


Figure 4.8. ACMP Input Front-end

The `initAcmp()` function in `opamp_acmp.c` routes the OPA0 output (V_{OUT0}) to ACMP0 positive input and OPA1 output (V_{OUT1}) to ACMP0 negative input. This example configures OPA0 and OPA1 as non-inverting amplifier with R2/R1 = 1/3.

Press "h" to run this example. The device stays in EM3 and wakes up with the ACMP falling edge interrupt (V_{OUT0} drops below V_{OUT1}).

```
OPA as ACMP input front-end at EM3
Reset the STK/WSTK to select another example

Device wakes up by ACMP falling edge interrupt
Reset the STK/WSTK to select another example
```

Table 4.8. Starter Kit/Radio Board for Opamp as ACMP Input Front-end Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A	EFM32TG11B520F128GM80	V_{IN1}	PA14 — APORT1XCH14	J101 pin 5
		V_{IN2}	PD6 — OPA1_POS	Expansion header pin 16
SLSTK3701A	EFM32GG11B820F2048GL192	V_{IN1}	PE10 — APORT3XCH10	Expansion header pin 4
		V_{IN2}	PE11 — APORT4XCH11	Expansion header pin 6
SLSTK3402A	EFM32PG12B500F1024GL125	V_{IN1}	PC9 — APORT2XCH9	Expansion header pin 3
		V_{IN2}	PC10 — APORT1XCH10	Expansion header pin 16
BRD4161A	EFR32MG12P432F1024GL125	V_{IN1}	PC9 — APORT2XCH9	Expansion header pin 13
		V_{IN2}	PC10 — APORT1XCH10	Expansion header pin 15
• BRD4159A • BRD4257A	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN1}	PC9 — APORT2XCH9	Expansion header pin 10
		V_{IN2}	PC10 — APORT1XCH10	Expansion header pin 15

4.10 Opamp as ADC Input Front-end

All opamps can route their outputs directly to the ADCn input as shown in Figure 4.9 ADC Input Front-end on page 28. The opamp output selection is done by POSSEL bitfield in ADCn_SINGLECTRL register. The ADCn can use opamps as its input front-end. When OPA0/1 is used as the VDACn output stage, the VDACn can also route its output VDACn_OUT0/1 to the ADCn input.

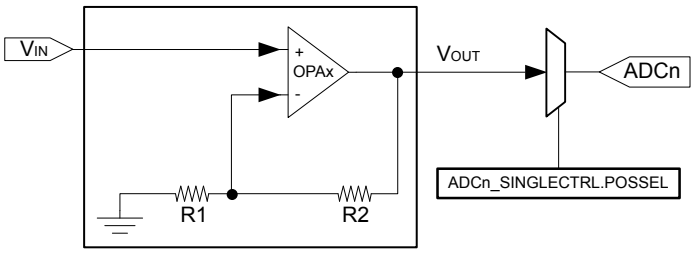


Figure 4.9. ADC Input Front-end

This example configures OPA0 as non-inverting amplifier with R2/R1 = 1/3. The `initOpaPrs()` function in `opamp.c` is used to configure the opamp PRS mode, input channel, out mode, and output channel (see 3.8 Enable Source and PRS Output).

```
void initOpaPrs(OPAMP_TypeDef opaSelect,                // OPAx - OPA0/1/2/3
                OPAMP_PrsMode_TypeDef opaPrsMode,       // PULSED or TIMED mode
                OPAMP_PrsSel_TypeDef opaPrsInCh,        // PRS input channel
                OPAMP_PrsOut_TypeDef opaPrsOut,         // WARM or OUTVALID output to PRS
                uint32_t opaPrsOutCh)                   // PRS output channel
```

Example of `initOpaPrs()` usage on OPA0:

```
initOpaPrs(OPA0, opaPrsModeTimed, opaPrsSelCh1, opaPrsOutOutValid, 2);
```

The OPA0 PRS configurations for this example are listed below.

- PRS enable mode — TIMED
- PRS input channel — 1 (from LETIMER0)
- PRS output mode — OUTVALID
- PRS output channel — 2 (to ADC0)

This example configures the LETIMER0 as a PRS producer (`initLetimer()` in `opamp_adc.c`) to enable the OPA0 at 16 Hz. The ADC single conversion (`initAdc()` in `opamp_adc.c`) is triggered by OPA0 PRS output when the OPA0 output is valid (see Figure 4.10 Timing Diagram of PRS channels on page 29). The LDMA is set up to transfer (`initLdma()` in `opamp_adc.c`) the ADC result from single conversion FIFO to RAM buffer in EM2 while the ADC continues with the next conversion.

The ADC0 configurations for this example are listed below.

- PRS input channel — 2 (from OPA0)
- Reference — AVDD
- Input — V_{OUT} of OPA0 (single conversion)
- Single FIFO DVL# — 3 (four samples to generate DMA request)
- ADC_CLK — 4 MHz AUXHFRCO in ASYNC mode for EM2 operation
- adc_clk_sar — 1 MHz (PRESC in ADCn_CTRL register = 3) in ASYNC mode
- ADC acquisition time — 1 adc_clk_sar cycle
- ADC conversion resolution — 12-bit
- ADC sampling frequency — 16 Hz (triggers by OPA0 PRS in EM2)
- ADC DMA request — ADC SINGLE request (REQ)

The timing diagram of the PRS channels is shown in Figure 4.10 Timing Diagram of PRS channels on page 29.

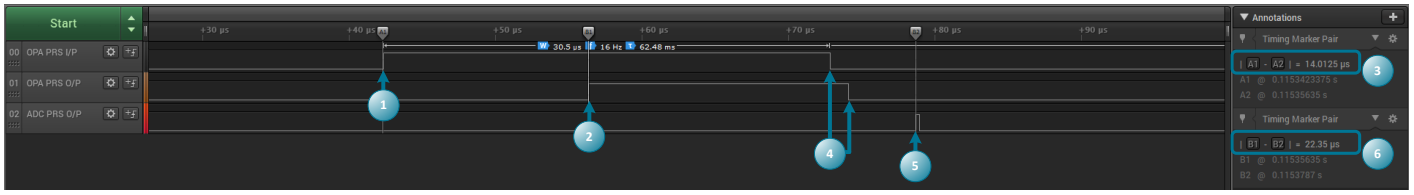


Figure 4.10. Timing Diagram of PRS channels

1. The LETIMER0 PRS output goes high (pulse width is about 30.5 µs — one LFXO or LFRCO cycle) to enable OPA0
2. The OPA0 PRS output goes high to trigger ADC conversion when OPA0 output is valid (PRS output mode = OUTVALID)
3. The time (A1 - A2) between positive edges of LETIMER0 PRS and OPA0 PRS pulses, which is equal to STARTDLY + WARMUP-TIME + SETTLETIME (see 3.7 Timing)
4. The LETIMER0 PRS output goes low to disable OPA0 (PRS enable mode = TIMED)
5. The ADC0 PRS output goes high (pulse width is about 0.25 µs — one ADC_CLK cycle) to signal end of ADC single conversion
6. The time (B1 - B2) between positive edges of OPA0 PRS and ADC0 PRS pulses

```
B1 - B2 = ADC request a wakeup to start of the peripheral clocks + 4 MHz AUXHFRCO start-up time
          + ADC warm-up time + ADC acquisition time + (ADC conversion resolution + 1) x T_adc_clk_sar
        = 2 µs + 2.5 µs + 5 µs + 1 µs + (12 + 1) x 1 µs
        = 23.5 µs
```

Note: The ADC acquisition phase must complete before the opamp is disabled.

Press "i" to run this example. The device stays in EM2 and wakes up with the LDMA DONE (16 samples — one second) interrupt to print out the latest ADC input voltage on the host computer.

```
OPA as ADC input front-end at EM2
Reset the STK/WSTK to select another example

ADC input voltage: 2076 mV
Reset the STK/WSTK to select another example
```

The average current consumption (measured by Energy Profiler in Simplicity Studio) of this example on SLSTK3402A is about 22.14 µA as shown in Figure 4.11 Current Consumption with ADC Triggered by OPA PRS on page 29.

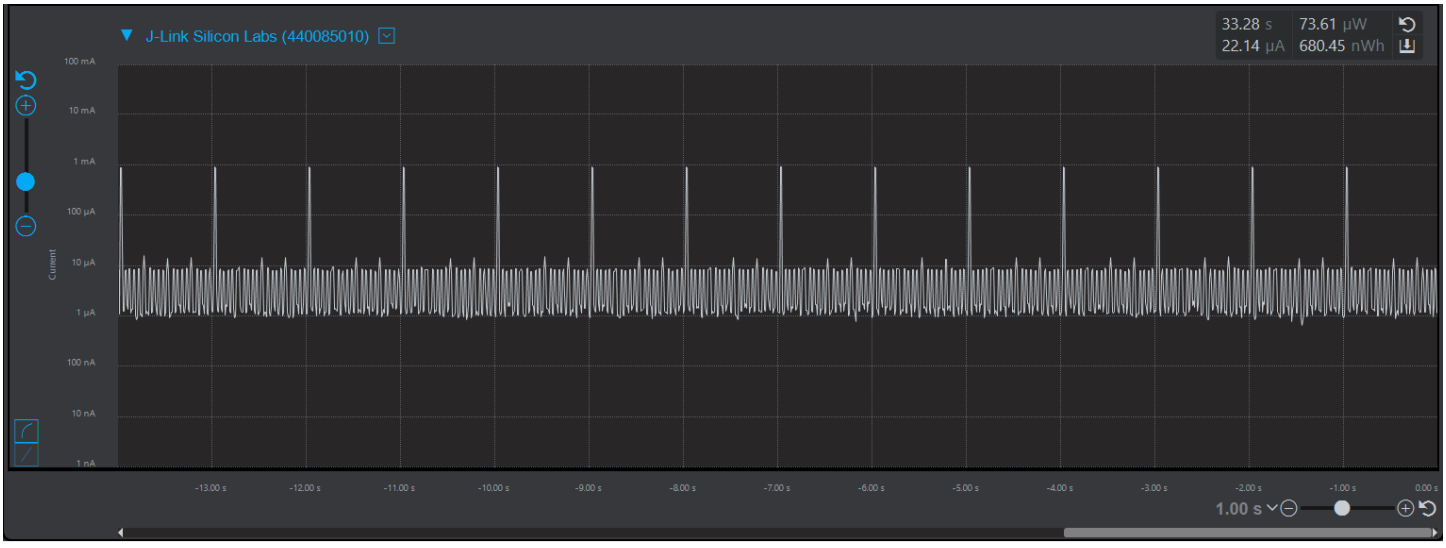


Figure 4.11. Current Consumption with ADC Triggered by OPA PRS

The LETIMER0 PRS output can directly trigger the ADC by keeping OPA0 active (set the `USE_OPA_PRS` define in `opamp_adc.h` to 0) and the average current consumption on SLSTK3402A will increase to about 115.81 µA as shown in Figure 4.12 Current Consumption with ADC Triggered by LETIMER0 PRS on page 30.

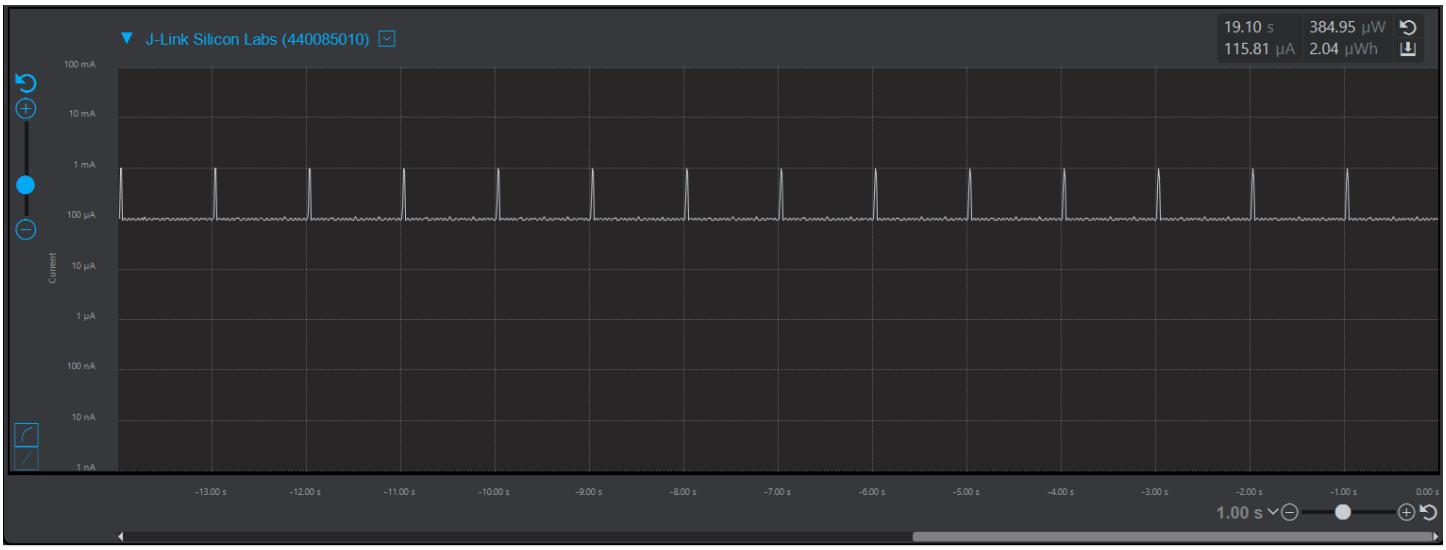


Figure 4.12. Current Consumption with ADC Triggered by LETIMER0 PRS

Table 4.9. Starter Kit/Radio Board for Opamp as ADC Input Front-end Example

Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A (OPA0)	EFM32TG11B520F128GM80	V_{IN}	PA14 — APORT1XCH14	J101 pin 5
SLSTK3701A (OPA0)	EFM32GG11B820F2048GL192	V_{IN}	PE10 — APORT3XCH10	Expansion header pin 4
SLSTK3402A (OPA0)	EFM32PG12B500F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 3
BRD4161A (OPA0)	EFR32MG12P432F1024GL125	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 13
• BRD4159A (OPA0) • BRD4257A (OPA0)	• EFR32MG13P632F512GM48 • EFR32FG14P233F256GM48	V_{IN}	PC9 — APORT2XCH9	Expansion header pin 10

4.11 Three Opamp Differential Amplifier

This configuration must use OPA0, OPA1, and OPA2. OPA0 and OPA1 are just used as voltage followers, and their outputs are routed to the non-inverting and inverting input of OPA2. Negative feedback is enabled in OPA2, and the non-inverting input uses the OPA0 resistor ladder. The OPA0 resistor ladder input is connected to ground. See figure Figure 4.13 Three Opamp Differential Amplifier on page 31 for an overview.

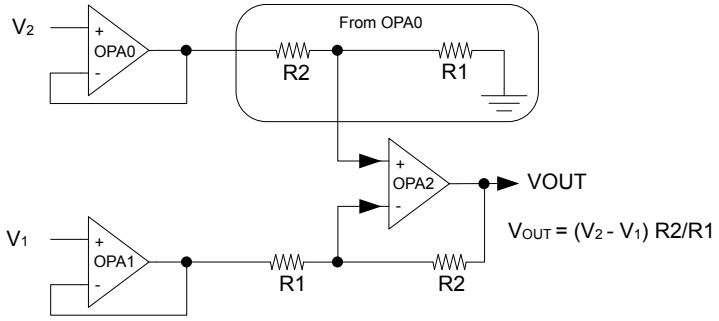


Figure 4.13. Three Opamp Differential Amplifier

There are limited gain selections available when R2 in OPA0 matches with R1 in OPA2, and R1 in OPA0 matches with R2 in OPA2. The available configurations can be found in Table 4.10 Three Opamp Differential Amplifier Gain Programming on page 31. This example uses R2 = 3 x R1 for OPA0 RESSEL and R2 = 1/3 x R1 for OPA2SEL so $V_{OUT} = (V_2 - V_1)$ x 1/3.

Table 4.10. Three Opamp Differential Amplifier Gain Programming

OPA0 RESEL	OPA2 RESSEL	Gain
4 (R2 = 3 x R1)	0 (R2 = 1/3 x R1)	1/3
1 (R2 = R1)	1 (R2 = R1)	1
0 (R2 = 1/3 x R1)	4 (R2 = 3 x R1)	3

The `opaThreeDiffAmp()` function in `opamp.c` is used to configure the three opamp differential amplifier.

```
void opaThreeDiffAmp(OPAMP_PosSel_TypeDef posInput0,    // V2
                     OPAMP_PosSel_TypeDef posInput1,    // V1
                     OPAMP_ResSel_TypeDef resSel0,      // OPA0 R2/R1
                     OPAMP_ResSel_TypeDef resSel2,      // OPA2 R2/R1
                     OPAMP_OutMode_TypeDef opaOutput,   // VOUT
                     uint32_t altOutMask)               // Enable bit mask if VOUT is alternate o/p
```

Example of `opaThreeDiffAmp()` usage on SLSTK3701A:

```
opaThreeDiffAmp(opaPosSelAPORT3XCH10, opaPosSelAPORT4XCH11, opaResSelR2eq3R1, opaResSelR2eq0_33R1,
                opaOutModeAPORT4YCH12, 0);
```

Press "j" to run this example and the device stays in EM3.

```
Three OPA differential amplifier at EM3
Reset the STK/WSTK to select another example
```

Table 4.11. Starter Kit/Radio Board for Three Opamp Differential Amplifier Example

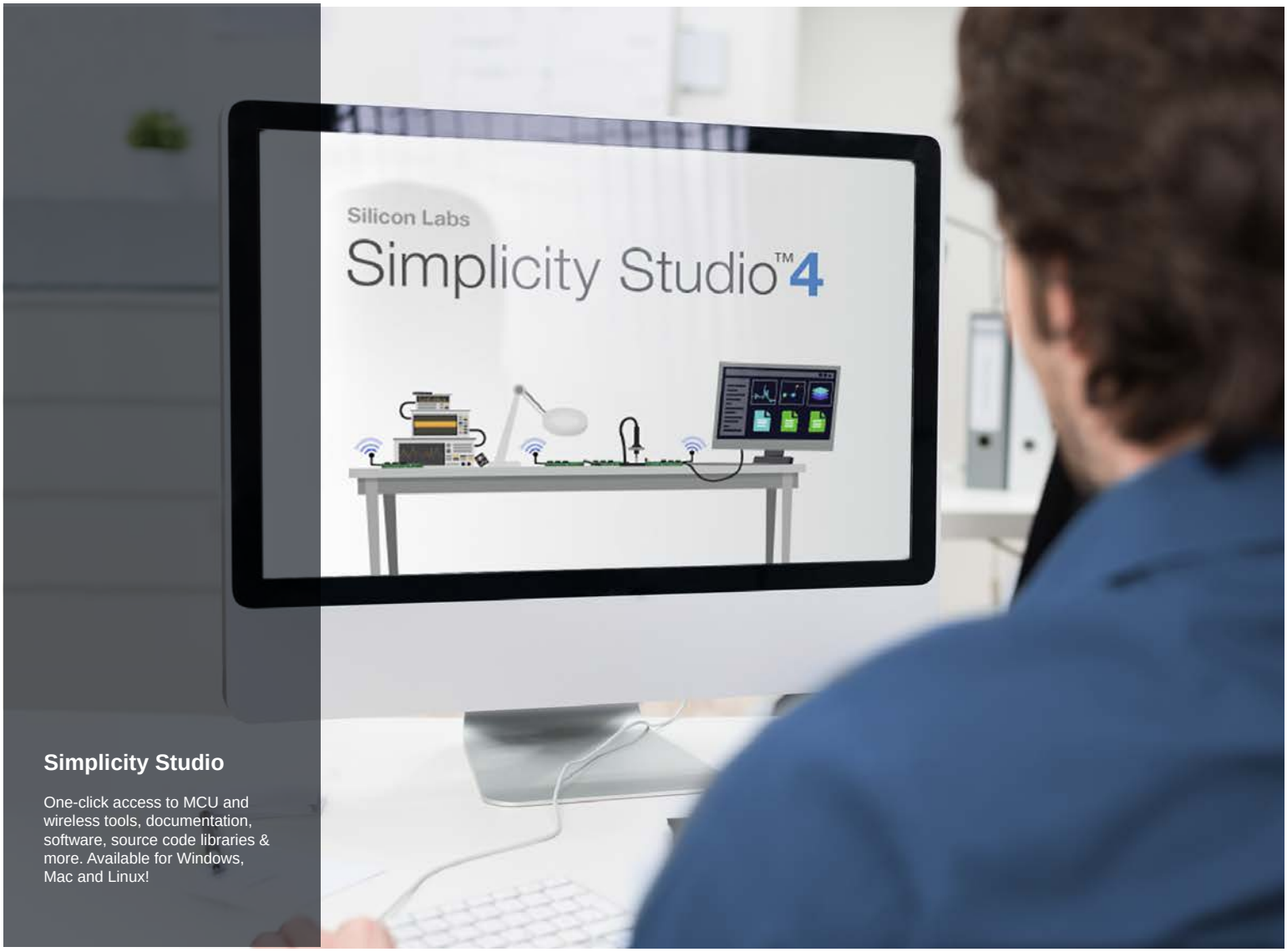
Starter Kit/Radio Board	Device	Signal	Pin	Kit Connection
SLSTK3301A	EFM32TG11B520F128GM80	V_2	PA14 — APORT1XCH14	J101 pin 5
		V_1	PD6 — OPA1_P	Expansion header pin 16
		V_{OUT}	PD5 — OPA2_OUT	Expansion header pin 11
SLSTK3701A	EFM32GG11B820F2048GL192	V_2	PE10 — APORT3XCH10	Expansion header pin 4
		V_1	PE11 — APORT4XCH11	Expansion header pin 6
		V_{OUT}	PE12 — APORT4YCH12	Expansion header pin 8
SLSTK3402A	EFM32PG12B500F1024GL125	V_2	PC9 — APORT2XCH9	Expansion header pin 3
		V_1	PC10 — APORT1XCH10	Expansion header pin 16
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 15
BRD4161A	EFR32MG12P432F1024GL125	V_2	PC9 — APORT2XCH9	Expansion header pin 13
		V_1	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16
BRD4159A	EFR32MG13P632F512GM48	V_2	PC9 — APORT2XCH9	Expansion header pin 10
		V_1	PC10 — APORT1XCH10	Expansion header pin 15
		V_{OUT}	PC11 — APORT1YCH11	Expansion header pin 16

5. Revision History

Revision 0.1

August, 2019

- Initial Revision



Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com