TB3286

Getting Started with Analog Signal Conditioning (OPAMP)

Introduction

Authors: Radu Toma, Martin Mostad, Microchip Technology Inc.

The Analog Signal Conditioning (OPAMP) peripheral features up to three internal operational amplifiers (op amps). It can help reduce or eliminate the need for external/discrete op amps in electronic designs, thus potentially decreasing the bill of materials. The main purpose of op amps is to condition the analog signals before the acquisition (and further digital processing) in a microcontroller or to provide the necessary output drive in control applications.

This technical brief describes how the analog signal conditioning block (OPAMP) part of the **AVR® DB MCU** devices works. It starts by describing the simplest configuration upon which more complex ones are built. The topologies of interest are as follows:

- **Op amp connected directly to pins:**
  The simplest and most basic configuration offering the highest degree of flexibility, with external connections and components.
- **Voltage Follower or Unity Gain Buffer:**
  A common configuration for converting a high impedance input to a low impedance output.
- **Non-Inverting Programmable Gain Amplifier:**
  Programmable gain signal amplification via the internal feedback resistor network.
- **Differential Amplifier using two op amps:**
  Differential input voltage amplification with a rejection of the common-mode voltage.
- **Instrumentation Amplifier using three op amps:**
  Differential signal amplification with high input and low output impedance.

Note: The code examples are designed for AVR128DB48 Curiosity Nano Evaluation Kit (EV35L43A) and are available on GitHub. There are stand-alone code examples for Atmel Studio, MPLAB® X IDE as well as Atmel START and MCC examples. Only the code used for the stand-alone examples are described in detail in this technical brief, but all examples function the same. For an introduction to the OPAMP module in Atmel START and MCC, see 3. Atmel START and 4. MPLAB® X MCC respectively.

View Code Example on GitHub
Click to browse repository

Hardware Configuration

Code examples are developed for the AVR128DB48 Curiosity Nano Evaluation Kit (EV35L43A).

Figure 1. AVR128DB48 Curiosity Nano

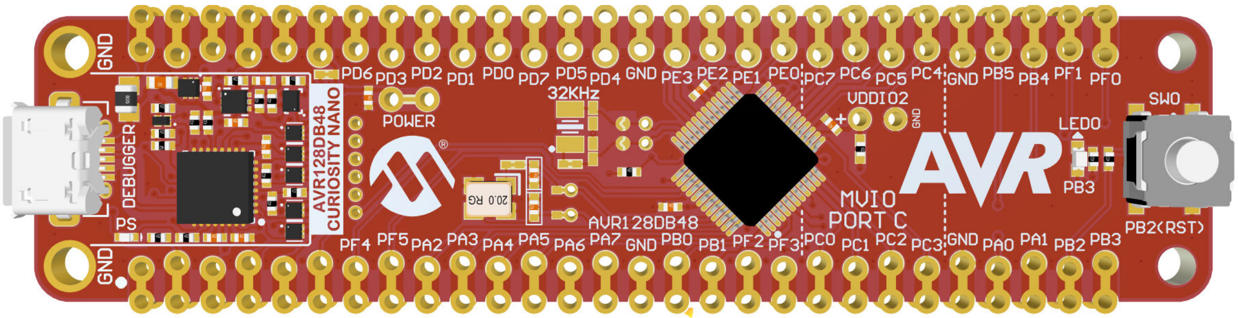


Table of Contents

1. Overview

The Analog Signal Conditioning (OPAMP) peripheral features one, two or three operational amplifiers (op amps), designated OPn where n is zero, one or two. These op amps are implemented with a flexible connection scheme using analog multiplexers and resistor ladders. This allows a large number of analog signal conditioning configurations to be achieved, many of which require no external components. A multiplexer at the non-inverting (+) input of each op amp allows connection to either an external pin, a wiper position from a resistor ladder, a DAC output, ground, or $V_{DD}/2$. A second multiplexer at the inverting (-) input of each op amp allows connection to either an external pin, a wiper position from a resistor ladder, the output of the op amp, or DAC output. Three more multiplexers connected to each resistor ladder provide additional configuration flexibility. Two of these multiplexers select the top and bottom connections to the resistor ladder, and the third controls the wiper position.

Figure 1-1. Block Diagram

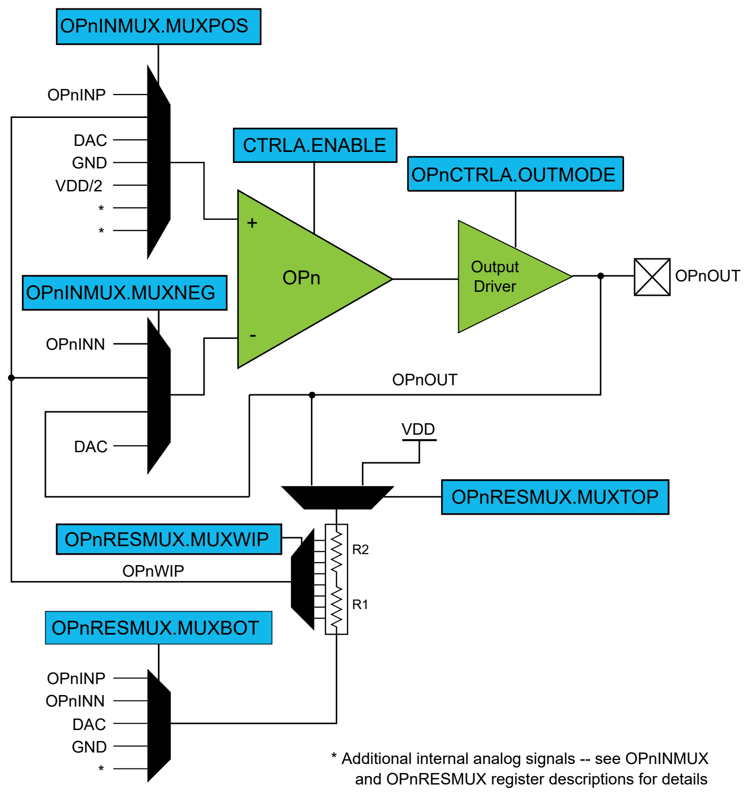


Table 1-1. Signal Description

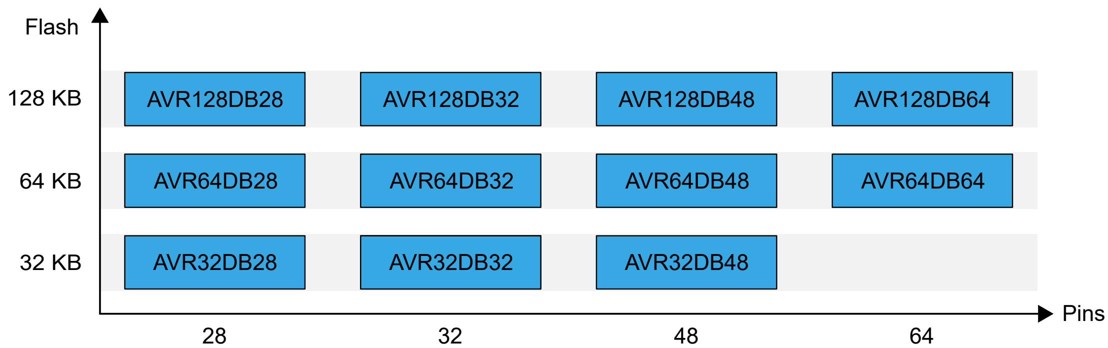
Signal Name	Type	Description
OPnINP	Analog input	Non-inverting (+) input pin for OPn
OPnINN	Analog input	Inverting (-) input pin for OPn
OPnOUT	Analog output	Output from OPn

2. Relevant Devices

This section lists the relevant devices for this document. The following figures show the different family devices, laying out pin count variants and memory sizes:

- Vertical migration upwards is possible without code modification, as these devices are pin-compatible and provide the same or more features
- Horizontal migration to the left reduces the pin count and, therefore, the available features
- Devices with different Flash memory sizes typically also have different SRAM and EEPROM

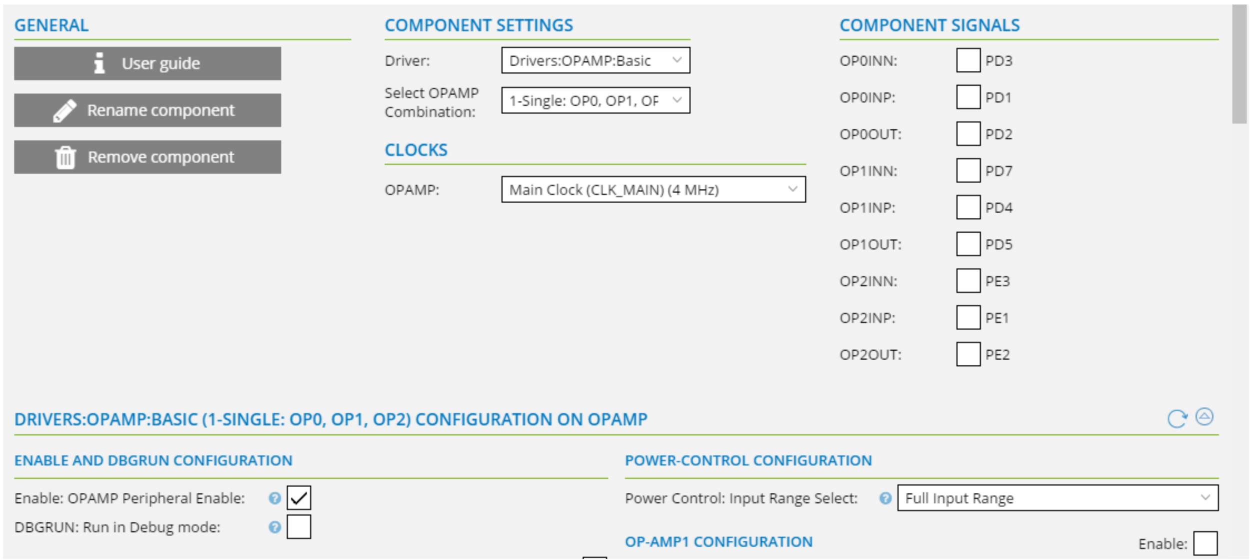
Figure 2-1. AVR® DB Family Overview



3. Atmel START

Atmel START comes with a driver for the OPAMP module allowing for easy configuration of the module. The Atmel START module comes with nine predefined configurations as well as a custom mode.

Figure 3-1. Atmel START Initial Configurations



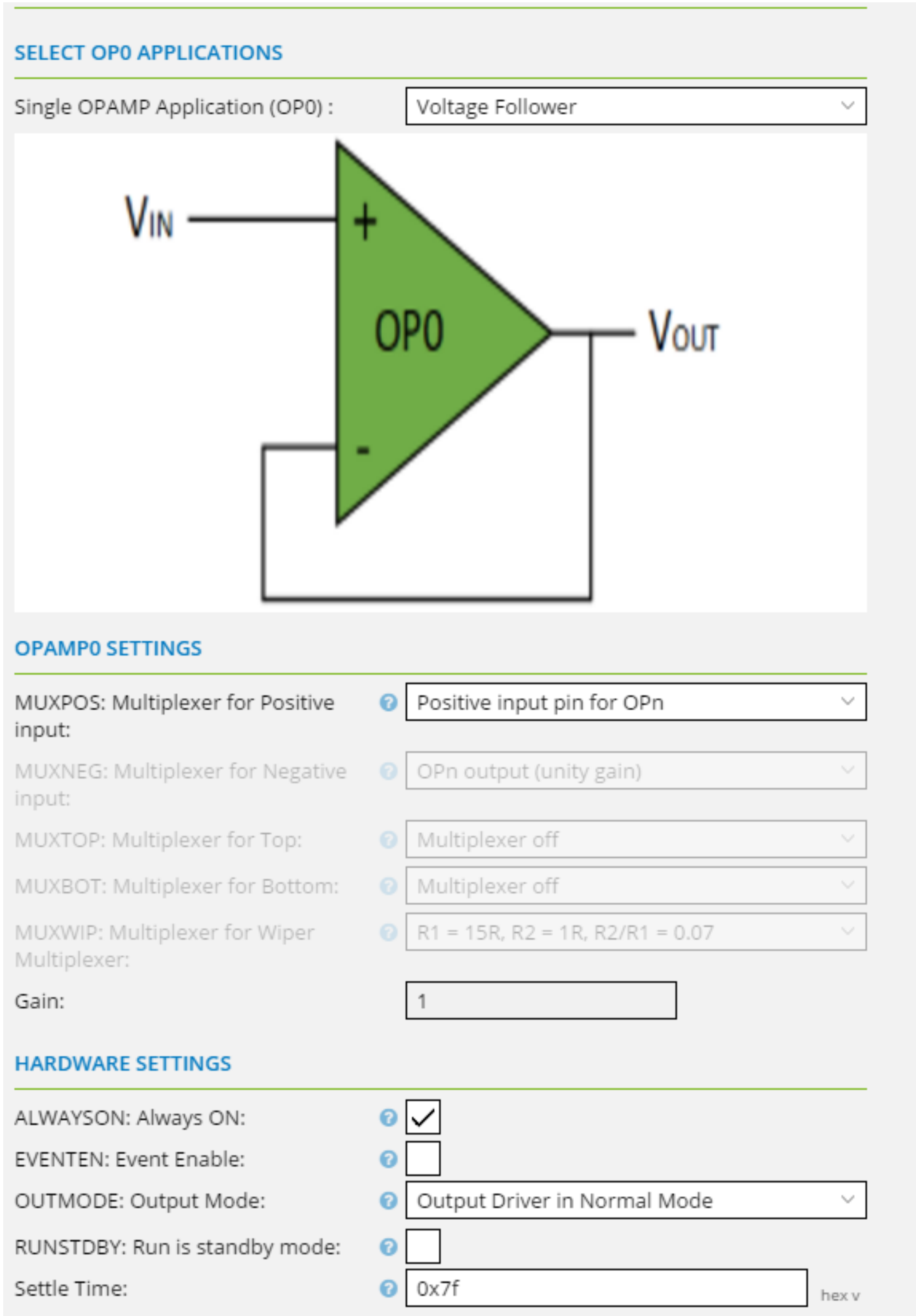
At the top of the module, as seen in Figure 3-1, several global options that can be set for the module, such as enabling the module, enabling the module in debug mode, and selecting the input range. Under “Component Signals”, the input and output pins can be selected. Selecting the pins will turn off the digital input buffer for the pins. It is therefore recommended to select all used pins. The “Select OPAMP Combination” option is used to select which op amps are configured separately and which op amps are combined to create a multi op amp configuration. All op amps can be configured to any of the single op amp configurations, but some of the multi op amp configurations are only available using specific op amps. For an overview of possible combinations, see Table 3-1.

Table 3-1. Multi Op Amp Configurations

Op Amp Combinations	Posible Configurations
OP1-OP2	Differential Amplifier, Cascaded (Two) Non-Inverting PGA, Cascading (Two) Inverting PGA
OP2-OP1	Differential Amplifier, Cascaded (Two) Non-Inverting PGA, Cascading (Tow) Inverting PGA
OP2-OP0	Differential Amplifier, Cascading (Two) Inverting PGA
OP0-OP1-OP2	Instrumentation Amplifier, Cascaded (Three) Non-Inverting PGA, Cascaded (Three) Inverting PGA
OP1-OP2-OP0	Instrumentation Amplifier, Cascaded (Three) Inverting PGA
OP2-OP0-OP1	Instrumentation Amplifier, Cascaded (Three) Inverting PGA

After selecting the global options for the OPAMP module, the settings for the individual op amps can be set. The first option is to select which configuration is desired. For each configuration, there is an accompanying picture to show how the configuration looks. The OPAMPn settings will have different fields grayed out depending on the configurations. The grayed-out field cannot be changed as that would break the configuration, but they are visible to show how the op amp is configured. The other field can be changed as normal and typically change the input or the gain of the configuration.

Figure 3-2. Atmel START Op Amp Configuration

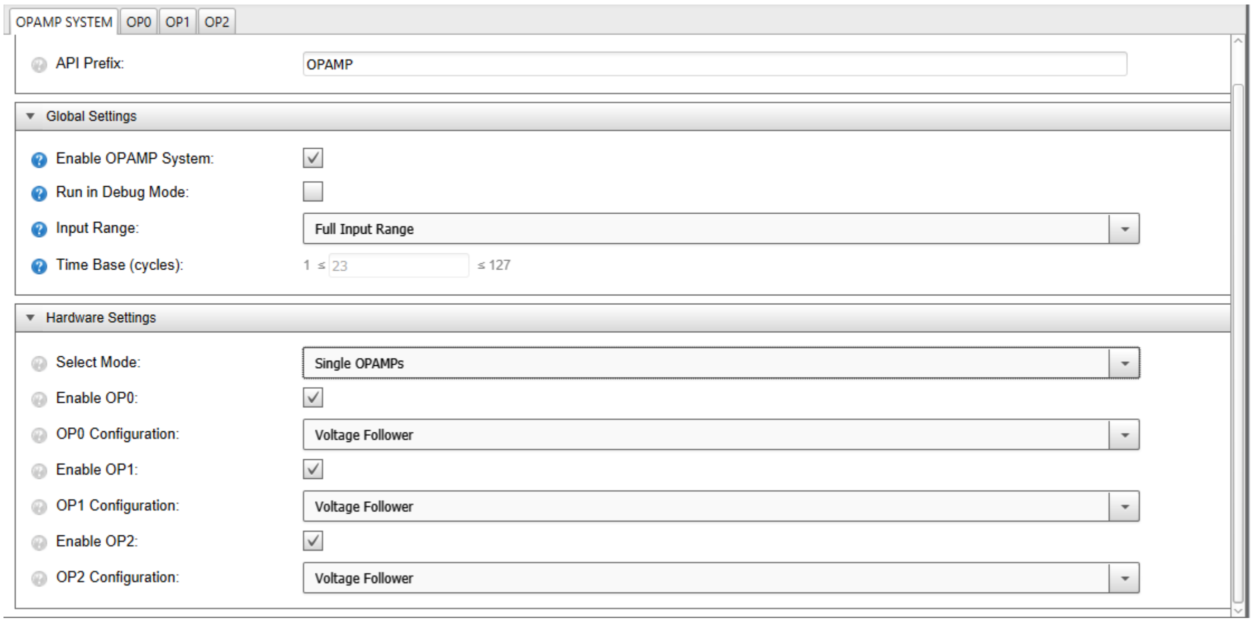


The last options ("HARDWARE SETTINGS") are hardware-specific to each op amp. By default, they are configured as always on and in normal output mode. This way, the op amp works as soon as it is initialized. The settle time is set to the maximum, as this is the srecommended value for an unknown load.

4. MPLAB® X MCC

MPLAB® X MCC comes with a driver for the OPAMP module allowing for easy configuration of the module. The MCC module comes with nine predefined configurations as well as a custom mode.

Figure 4-1. MCC Global and Hardware Settings



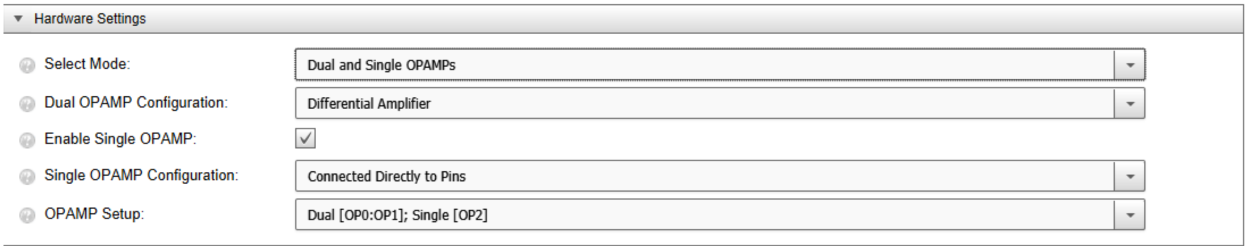
Under the “OPAMP SYSTEM” banner, as seen in Figure 4-1, there are several global options that can be set for the module, such as enabling the module, enabling the module in debug mode, and selecting the input range. Under “Hardware Settings”, the select mode option is used to select if the op amp should be configured separately or be combined into multi op amp configurations. All op amps can be configured to any of the single op amp configurations, but some of the multi op amp configurations are only available using specific op amps. For an overview of possible combinations, see Table 4-1.

Table 4-1. Multi Op Amp Configurations

Op Amp Combinations	Posible Configurations
OP1-OP2	Differential Amplifier, Cascaded (Two) Non-Inverting PGA, Cascading (Two) Inverting PGA
OP2-OP1	Differential Amplifier, Cascaded (Two) Non-Inverting PGA, Cascading (Tow) Inverting PGA
OP2-OP0	Differential Amplifier, Cascading (Two) Inverting PGA
OP0-OP1-OP2	Instrumentation Amplifier, Cascaded (Three) Non-Inverting PGA, Cascaded (Three) Inverting PGA
OP1-OP2-OP0	Instrumentation Amplifier, Cascaded (Three) Inverting PGA
OP2-OP0-OP1	Instrumentation Amplifier, Cascaded (Three) Inverting PGA

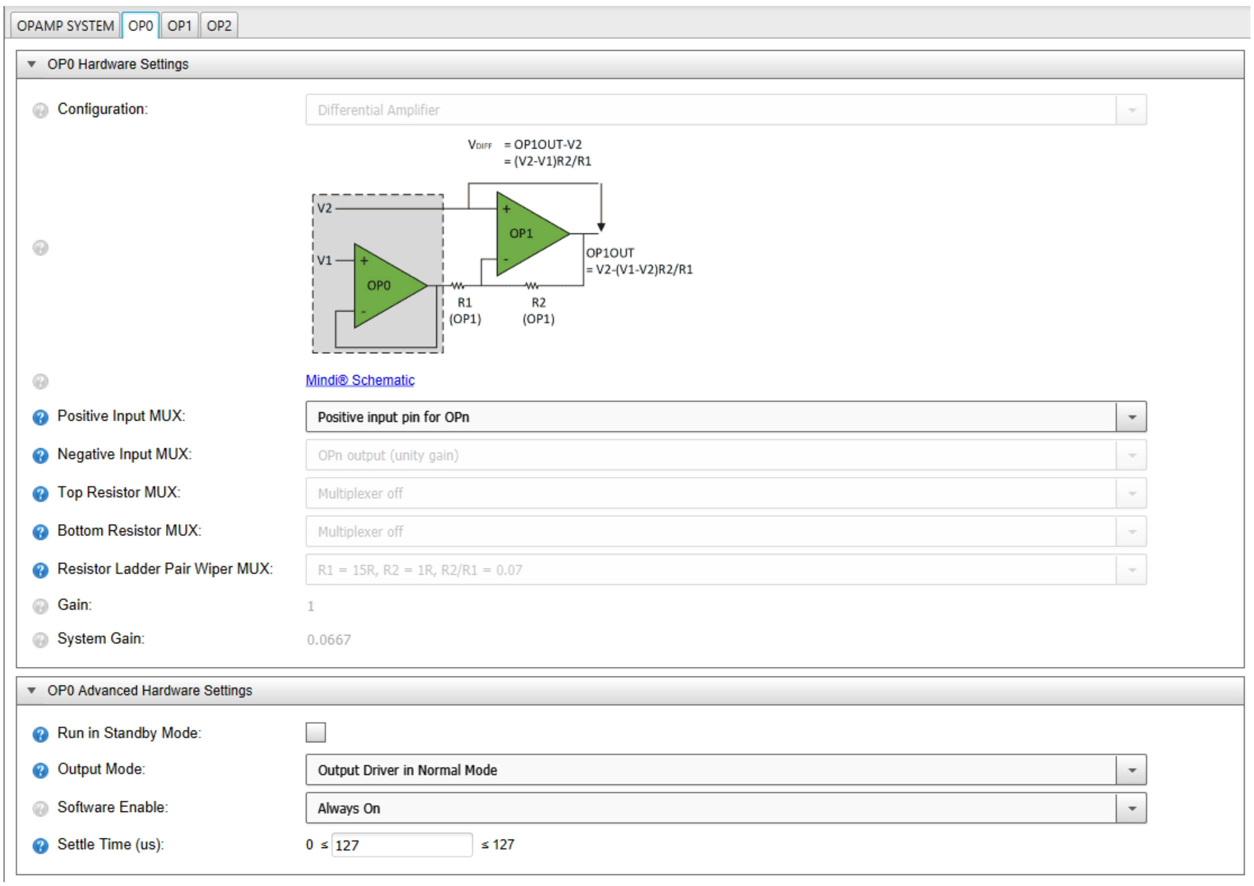
If the *Single OPAMPs* option is selected, each op amp can be enabled and configured separately, as seen in Figure 4-1. If the *Dual and Single OPAMPs* or *Triple OPAMPs* option is selected, the configuration of the dual or triple configuration can be selected. The configuration of the single op amp, if any, and how the op amps should be connected, is shown in Figure 4-2.

Figure 4-2. Dual and Single OPAMPs Configurations



After selecting the global options and hardware settings, there is one banner for each enabled op amp where the op amp specific options can be selected. For each configuration, there is an accompanying picture to show how the configuration looks. The “OP0 Hardware Settings” will have different fields grayed out depending on the configurations. The grayed-out field cannot be changed as that would break the configuration, but they are visible to show how the op amp is configured. The other field can be changed as normal and typically change the input or the gain of the configuration.

Figure 4-3. OPn Hardware Settings



The last options are under “OPn Advanced Hardware” Settings. By default, they are configured as always on and in normal output mode. This way, the op amp works as soon as it is initialized. The settle time is set to the maximum, as this is the recommended value for an unknown load.

5. MPLAB® Mindi™ Analog Simulator

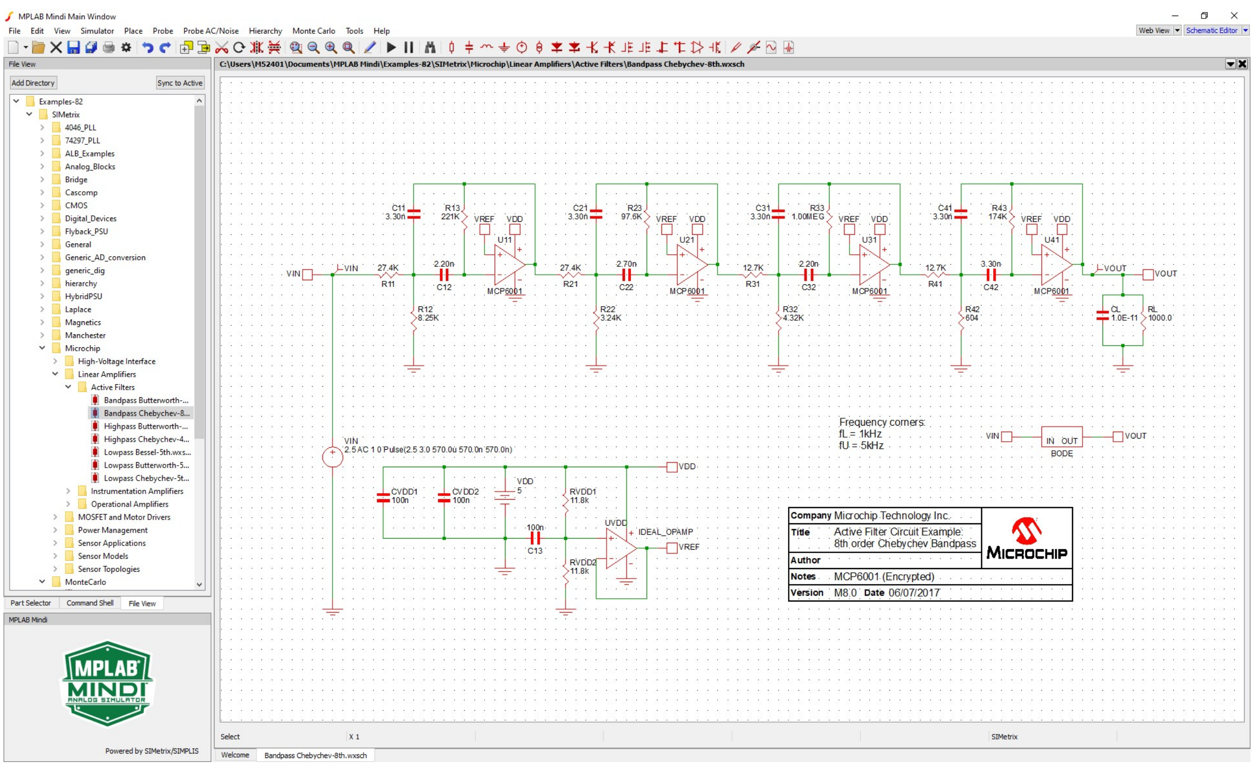
Simulating electronic circuits can save development time and resources by reducing design iterations. Costly design errors can be found at an early stage and corrected without much effort. Simulations also have an important learning aspect where the operation of a circuit can quickly be characterized and understood.

The MPLAB® Mindi™ Analog Simulator is a comprehensive tool for analog circuit design and analysis. The tool uses a SIMetrix/SIMPLIS simulation environment, that can cover a very wide set of possible simulation needs. It has an easy-to-use interface, fast simulation times, and an ever-growing library of models and application schematics. Available model libraries include: Operational amplifiers, active filter circuits, MOSFET and motor drivers, power modules, LED drivers, switching regulators, generic switch, and passive components.

MPLAB® Mindi™ installs and runs locally. Once downloaded, no internet connection is required, and the simulation run-time is not dependent on a remotely located server. The result is fast, accurate analog circuit simulations. Applications that can greatly benefit from MPLAB® Mindi™ include:

- Generation of BODE responses for active and passive filter systems
- Evaluation of transient responses to a wide variety of input conditions
- Generation of closed loop stability responses for control systems
- Verification of slew rates and drive strengths through power drive or signal conditioning chains
- Modeling the noise effects in signal conditioning or control systems

Figure 5-1. MPLAB® Mindi™ - Overview of Design Environment



The latest version of MPLAB® Mindi™ is available for download on the MPLAB® Mindi™ Analog Simulator product page.

Each described topology comes with a Mindi™ schematic and simulation, showcasing the configuration through a simple example. It also sets the premises for what is to be expected from the functional hardware configuration once the firmware has been programmed in the target device.

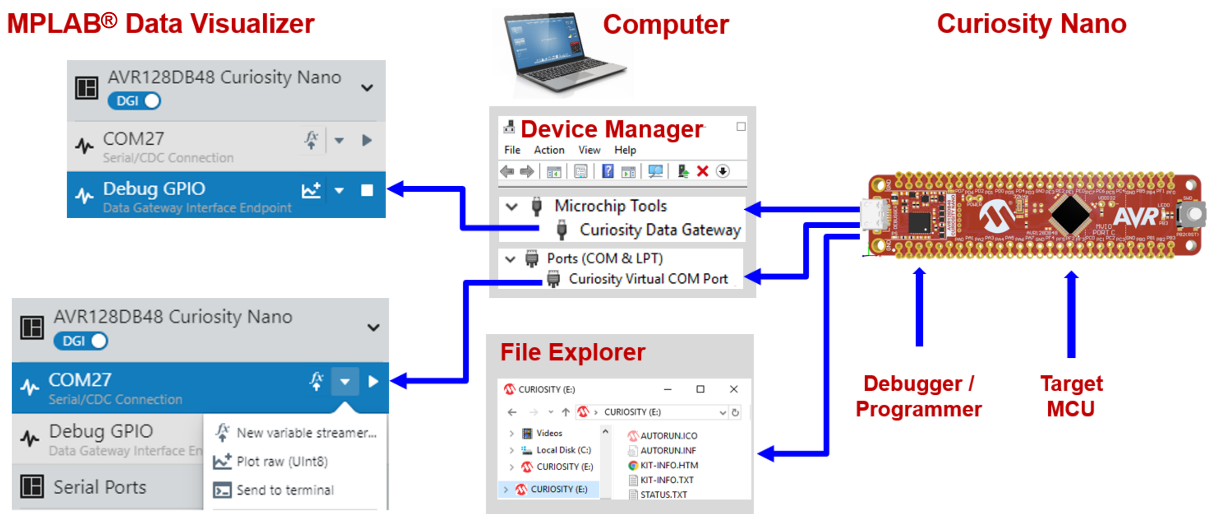
6. MPLAB® Data Visualizer

The MPLAB® *Data Visualizer* is a program used to process and visualize key data points, from a running embedded target, in real-time. The utility may be accessed as an MPLAB® X IDE plugin or as a standalone program. The latest version of MPLAB® *Data Visualizer* is available for download on the MPLAB® Data Visualizer product page.

Figure 6-1 outlines the main concepts and features available, such as:

- Capture data, streamed from a running embedded target, via virtual serial port (USB) or the Data Gateway Interface (DGI)
- Decode data fields at run-time using the Data Stream Protocol format
- Visualize the raw or decoded data in a Graph as a time series or display the data in a terminal
- Concurrently stream data and debug target code

Figure 6-1. MPLAB® Data Visualizer



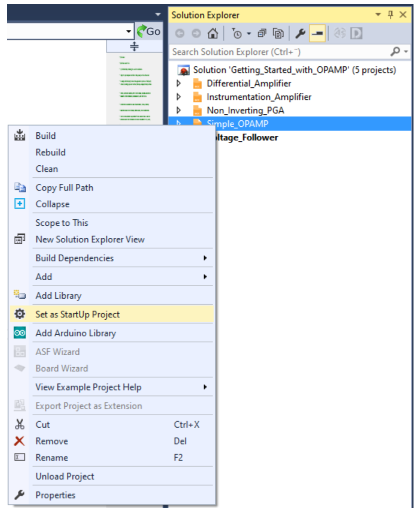
Each configuration is set up to send data over USART as a *Data Visualizer data stream*. The values of interest are the input and output signals. Both signals will be displayed in the main *Graph* area of the MPLAB® *Data Visualizer*. The interface and underlying settings are configured through a saved workspace available in a folder, named *data-visualizer*.

To get a workspace up and running, follow the steps below:

Note: There is no MPLAB® *Data Visualizer* workspace available for the *Simple OPAMP* project.

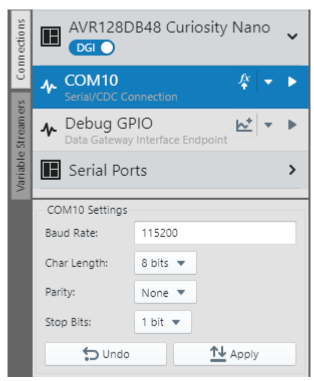
1. Open the Atmel Studio 7 solution *Getting_Started_with_OPAMP* by navigating to the GitHub repository and selecting *Getting_Started_with_OPAMP.atsln*. The solution contains separate projects for each configuration detailed in the present document.
2. Set the desired project as *StartUp Project*, by right clicking on the project of interest and from the pop-up menu select *Set as StartUp Project* option.

Figure 6-2. Atmel Studio 7 - Set Start-up Project



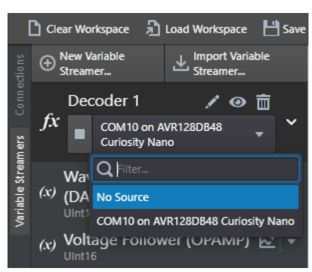
3. Build the solution by selecting *Build* → *Build Solution* or by pressing F7.
4. Program the AVR DB device by selecting *Debug* → *Start Without Debugging* from the top menu bar.
5. Open MPLAB® *Data Visualizer*.
6. Load the workspace. Press the *Load Workspace* button and add the workspace-file corresponding to the configuration, available in the GitHub repository. Two axes should appear in the graph.
7. Set the correct Serial Port settings. Choose the COM-port on the left hand side panel seen in Figure 6-3. Make sure the *Baud Rate* is 115200 and press the play button (*Start Streaming*).

Figure 6-3. MPLAB® Data Visualizer - Serial Port Settings



8. Ensure the COM port is selected as a source for the Variable Streamer Decoder.

Figure 6-4. MPLAB® Data Visualizer - Variable Streamers



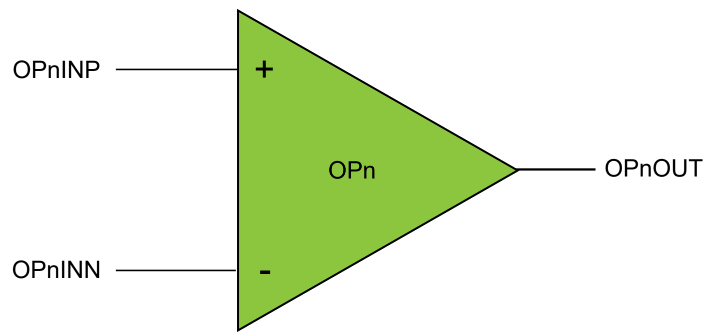
9. If successful, two graphs representing the input and output signals will be drawn on the screen. The Y-axis represents the magnitude of the signal in mV, while the X-axis represents elapsed time.

Note: For more information relating to MPLAB® *Data Visualizer*, consult the relevant documentation resources.

7. Op Amp Basic Configuration

Figure 7-1 displays an op amp connected directly to the pins of the device. The input and output of the op amp are not connected to the internal feedback resistor network.

Figure 7-1. Op Amp Connected Directly to I/O Pins



7.1 Use Cases

This configuration is useful for situations where the user desires to make all connections to other components externally. For specific cases when an analog comparison function is desired, use the separate Analog Comparator block, which is purposely built for such applications.

7.2 MPLAB® Mindi™ Model

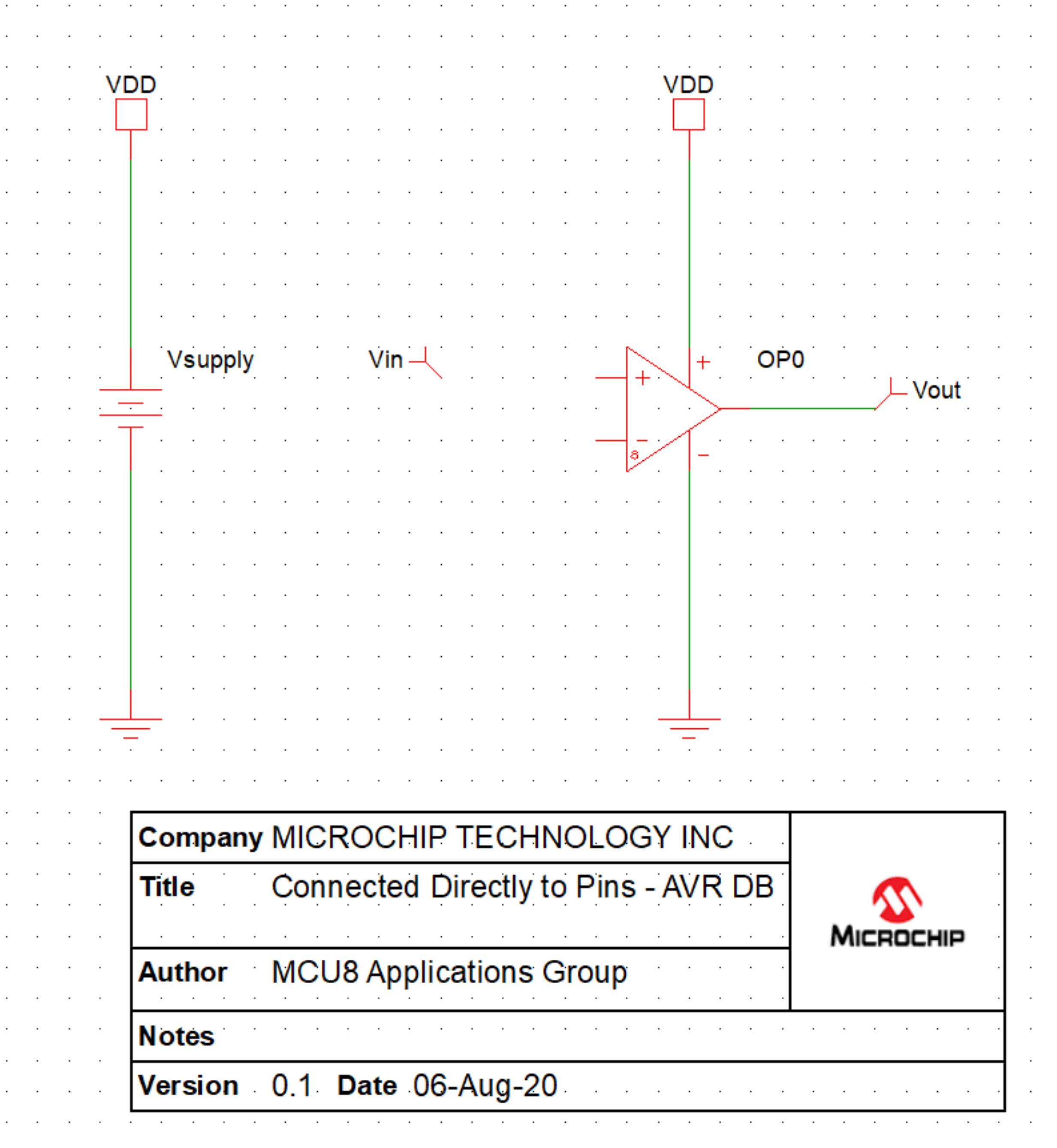
The op amps used in the MPLAB® Mindi™ are all created using the parameters found in the electrical DC and AC characteristics available in the AVR DB data sheet.

In MPLAB® Mindi™, open **opamp.wxsch** schematic from the GitHub repository.

View Code Examples on GitHub
Click to browse repositories

The circuit shown in Figure 7-2 contains the op amp without any external components allowing any standard op amp configuration to be achieved by connecting the appropriate external components.

Figure 7-2. Basic Configuration - Analog Simulation with MPLAB® Mindi™



Simulate by selecting; **Simulator** → **Run Schematic** or **press Function Key F9**.

7.3 Register Configuration

The Timebase register for the Analog Signal Conditioning (OPAMP) peripheral has to be configured first. The user must program the equivalent number of clock cycles that amount to 1 μs, so the contents are dependent on the operating CPU clock frequency.

Figure 7-3. OPAMP.TIMEBASE - Set TIMEBASE

Bit	7	6	5	4	3	2	1	0
		TIMEBASE[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	1

Bits 6:0 – TIMEBASE[6:0] Timebase
This bit field controls the maximum value of a counter that counts CLK_PER cycles to achieve a time interval equal to or larger than 1 µs. It should be written with one less than the number of CLK_PER cycles that are equal to or larger than 1 µs. This is used for internal timing of the warmup and settling times.

```
#define OPAMP_TIMEBASE_US     (ceil(F_CPU /1e6)-1)
OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;
```

In applications where a rail-to-rail input voltage range is not needed, the OPAMP peripheral may be configured to save power, by writing a '1' to the Input Range Select (IRSEL) bit in the Power Control (PWRCTRL) register. For the basic op amp applications in the technical brief the power saving option will not be used.

Figure 7-4. OPAMP.PWRCTRL - Set Input Range

Bit	7	6	5	4	3	2	1	0
								IRSEL
Access								R/W
Reset								0

Bit 0 – IRSEL Input Range Select
This bit selects the op amp input voltage range

Value	Description
0	The op amp input voltage range is rail-to-rail
1	The op amp input voltage range and power consumption are reduced. See the *Electrical Characteristics* section for more information

```
OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;
```

For a basic op amp configuration operation, the op amp is configured to be always on. It is assumed the ENABLE/DISABLE events are not used in this scenario to enable/disable the op amp. Similarly, the output driver is enabled by selecting the normal output mode. However, in a different application, the op amp could be enabled or disabled based on a certain event generator (TCA, TCB, TCD, RTC, PORT, CCL, etc.). One area where such functionality could be useful is power saving, where that op amp would be enabled only when needed, rather than having it on from the moment power is applied to the circuitry.

Figure 7-5. OPAMP.OPnCTRLA - Configure the Op Amp Control A

Bit	7	6	5	4	3	2	1	0
	RUNSTDBY				OUTMODE[1:0]		EVENTEN	ALWAYSON
Access	R/W				R/W	R/W	R/W	R/W
Reset	0				0	0	0	0

Bit 7 – RUNSTDBY Run in Standby Mode
This bit controls whether or not the op amp functions in Standby sleep mode.

Value	Description
0	OPn is disabled when in Standby sleep mode, and its output driver is disabled.
1	OPn will continue operating as configured in Standby sleep mode.

Bits 3:2 – OUTMODE[1:0] Output Mode
This bit field selects the output mode of the output driver.

Value	Name	Description
0x0	OFF	The output driver for OPn is disabled, but this can be overridden by the DRIVEn event.
0x1	NORMAL	The output driver for OPn is enabled in Normal mode.
0x2 - 0x3	-	Reserved

Bit 1 – EVENTEN Event Enable
This bit enables event reception and generation.

Value	Description
0	No events are enabled for OPn.
1	All events are enabled for OPn.

Bit 0 – ALWAYSON Always On
This bit controls whether the op amp is always on or not.

Value	Description
0	OPn is not always on, but can be enabled by the ENABLEn event and disabled by the DISABLEn event.
1	OPn is always on.

```
OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
```

For a basic op amp operation, the inputs and output of the op amp are connected directly to the pins of the device. The multiplexer settings required to achieve the basic op amp configuration are:

Table 7-1. Op Amp Connected Directly to Pins

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	INP	INN	OFF	WIP0	OFF

In the case of the basic op amp configuration, the resistor ladder configuration will remain `0x00`.

Figure 7-6. OPAMP.OPnRESMUX - Configure the Resistor Ladder Multiplexer

Bit	7	6	5	4	3	2	1	0
	MUXWIP[2:0]			MUXBOT[2:0]			MUXTOP[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:5 – MUXWIP[2:0] Multiplexer for Wiper
This bit field selects the resistor ladder wiper (potentiometer) position.

Value	Name	Description
0x0	WIP0	R1 = 15R, R2 = 1R
0x1	WIP1	R1 = 14R, R2 = 2R
0x2	WIP2	R1 = 12R, R2 = 4R
0x3	WIP3	R1 = 8R, R2 = 8R
0x4	WIP4	R1 = 6R, R2 = 10R
0x5	WIP5	R1 = 4R, R2 = 12R
0x6	WIP6	R1 = 2R, R2 = 14R
0x7	WIP7	R1 = 1R, R2 = 15R

Bits 4:2 – MUXBOT[2:0] Multiplexer for Bottom
This bit field selects the analog signal connected to the bottom resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	INP	Positive input pin for OPn
0x2	INN	Negative input pin for OPn
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x4	LINKOUT	OP[n-1] output (Setting only available for OP1)(1)
0x5	GND	Ground
Other	-	Reserved

Note: When selecting LINKOUT for OP0, MUXBOT is connected to the output of OP2.

Bits 1:0 – MUXTOP[1:0] Multiplexer for Top
This bit field selects the analog signal connected to the top resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	OUT	OPn output
0x2	VDD	V_{DD}
Other	-	Reserved

```
OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;
```

The negative and positive inputs to the op amp are connected straight to the I/O pins.

Figure 7-7. OPAMP.OPnINMUX - Configure the Input Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXNEG[2:0]				MUXPOS[2:0]		
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:4 – MUXNEG[2:0] Multiplexer for Negative Input
This bit field selects which analog signal is connected to the inverting (-) input of OPn.

Value	Name	Description
0x0	INN	Negative input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	OUT	OPn output (unity gain)
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
Other	-	Reserved

Bits 2:0 – MUXPOS[2:0] Multiplexer for Positive Input
This bit field selects which analog signal is connected to the non-inverting (+) input of OPn.

Value	Name	Description
0x0	INP	Positive input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x3	GND	Ground
0x4	VDDDIV2	$V_{DD/2}$
0x5	LINKOUT	OP[n-1] output (Setting only available for OP1 and OP2)
0x6	LINKWIP	Wiper from OP0's resistor ladder (Setting only available for OP2)
Other	-	Reserved

```
OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_INN_gc | OPAMP_OP0INMUX_MUXPOS_INP_gc;
```

The settling time depends on a variety of factors, including the load on the op amp, that may not be known until the later stages of design and development. If the settling time is unknown, the maximum value of '0x7F' (127 microseconds) should be written to the SETTLE bit field.

Figure 7-8. OPAMP.OPnSETTLE - Configure the Settle Time

Bit	7	6	5	4	3	2	1	0
		SETTLE[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 6:0 – SETTLE[6:0] Settle Time
This bit field specifies the number of microseconds allowed for the op amp output to settle. This value is used by an internal timer to determine when to generate the READYn event and set the SETTLED flag in the OPnSTATUS register.

The operational amplifier module is enabled by setting the ENABLE bit in the OPAMP.CTRLA register:

Figure 7-9. OPAMP.CTRLA - Enable OPAMP Peripheral

Bit	7	6	5	4	3	2	1	0
								ENABLE
Access								R/W
Reset								0

Bit 0 – ENABLE Enable OPAMP Peripheral
This bit controls whether the OPAMP peripheral is enabled or not.

Value	Description
0	The OPAMP peripheral is disabled.
1	The OPAMP peripheral is enabled.

```
OPAMP.CTRLA = OPAMP_ENABLE_bm;
```

A value of '0' of the SETTLED bit indicates that the settling time has elapsed:

Figure 7-10. OPAMP.OP0SnATUS - OPAMP Status

Bit	7	6	5	4	3	2	1	0
								SETTLED
Access								R
Reset								0

Bit 0 – SETTLED Op Amp has Settled
This bit is cleared when the op amp is waiting for settling time related to enabling or configuration changes.
This bit is set when the allowed settling time is finished.

```
while (OPAMP.OP0STATUS & OPAMP_SETTLED_bm)
  {
    ;
  }
```

Putting it all together, the basic op amp initialization code will look as follows:

```
void OPAMP0_init (void)
{
    /* Configure the Timebase */
    OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;

    /* Configure the voltage input range */
    OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;

    /* Configure the Op Amp n Control A */
    OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;

    /* Configure the Op Amp n Input Multiplexer */
    OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_INN_gc | OPAMP_OP0INMUX_MUXPOS_INP_gc;

    /* Configure the Op Amp n Resistor Wiper Multiplexer */
    OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;

    /* Configure the Op Amp n Settle Time*/
    OPAMP.OP0SETTLE = 0x7F;

    /* Enable OPAMP peripheral */
    OPAMP.CTRLA = OPAMP_ENABLE_bm;

    /* Wait for the operational amplifiers to settle */
    while (OPAMP.OP0STATUS & OPAMP_SETTLED_bm)
    {
        ;
    }
}
```

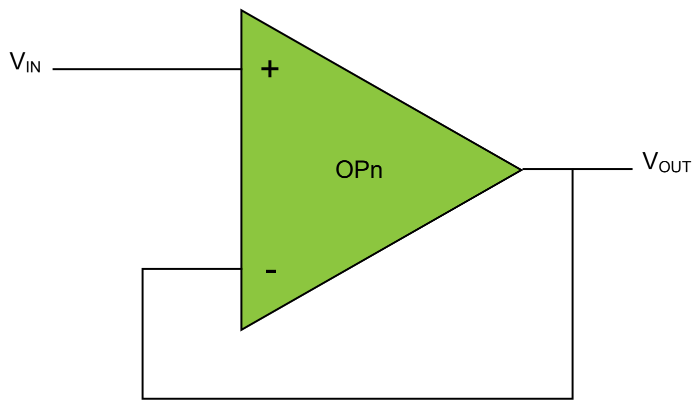
The code for this example is available in the **simple-opamp** folder in these github repositories
Note: This example is not available for Atmel START or MPLAB® X MCC

View Code Example on GitHub
Click to browse repository

8. Voltage Follower

Figure 8-1 displays an op amp in voltage follower configuration, also known as a unity-gain buffer or voltage buffer. The non-inverting (+) input is connected to a pin, and the output is connected to the inverting (-) input.

Figure 8-1. Voltage Follower



The output voltage is equal to the input voltage, giving the transfer function:

$$V_{OUT} = V_{IN}$$

The configuration does not provide any amplification thus the voltage gain is equal to `1`.

The voltage follower is mainly used for converting a high input impedance to a low output impedance. It is often found in the sample and hold circuits as an input to analog to digital converters (ADC) or buffers for logic circuits.

8.1 Use Case

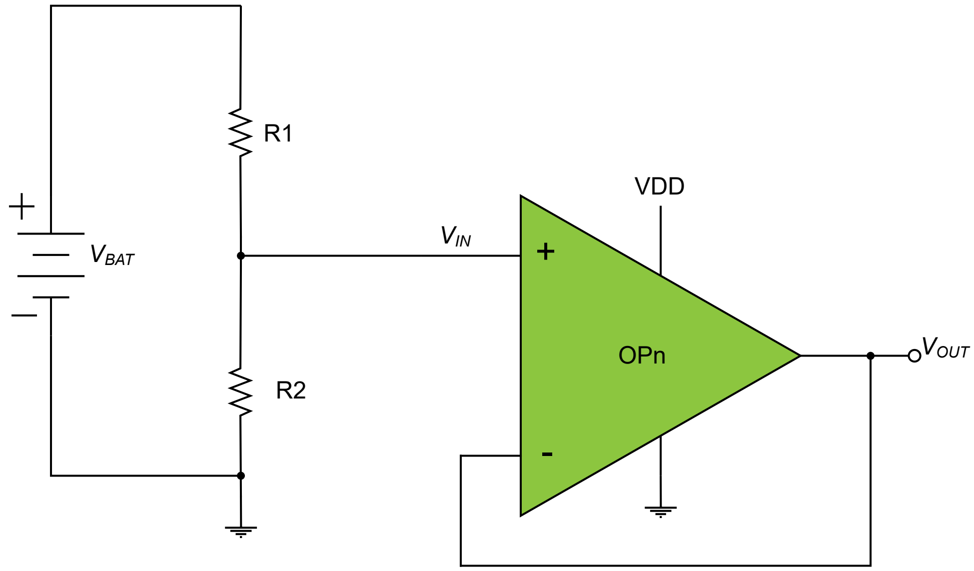
Battery Monitoring Systems (BMS) ICs are a common occurrence in battery-powered electronic products. At the very basic level, these components continuously measure the battery voltage and the load current. With this information, it is possible to calculate and inform the user when a battery needs charging or replacing before the device becomes unusable.

For cost-sensitive devices, a solution around a voltage divider and an operational amplifier could be the answer (see Figure 8-2). The input to the voltage buffer is a ratio of the battery voltage, while the output of the voltage buffer can be internally routed to an ADC for acquisition and further processing.

$$V_{OUT} = \frac{R_2}{R_1 + R_2} \times V_{BAT}$$

To minimize the current draw, the resistor values are chosen to be high, see Table 8-1 for typical values. Such a design choice usually poses a challenge for the microcontroller's ADC. Normally the ADC inputs are not buffered and the input current into the ADC will cause errors in the measurement. This is where the voltage follower/buffer comes in by adapting the high impedance of the voltage divider's output to the low impedance of the ADC input.

Figure 8-2. External Battery Voltage Buffer



In this design, the battery voltage has an independent voltage from the op amp's rail supply (which is usually the case). The battery voltage is brought within the amplifier's common-mode input voltage range by the R1 and R2 voltage divider. For the case of a 2S1P Li-On battery pack, the safe operating voltage level can be anywhere between 5V and 8.4V. However, the supply voltage for the MCU is regulated at 3V. A voltage divider with a ratio of 1:8.66 for the R2 and R1 brings the monitoring voltage between 0.5V and 0.87V (see the voltage transfer function above). The internal 1.024V band gap reference can be selected as the reference voltage for the ADC.

The important aspect is the selection of resistor values for the voltage divider to minimize current draw and reduce its impact on the battery lifetime. Table 8-1 briefly shows the effect of the voltage divider resistor values and does not take into account any potential loads, such as the operating current of the microcontroller . The resulting values for the battery lifetime assume an almost ideal environmental operation of the design, with the Li-On battery in the nominal voltage of 3.7V/cell and a self-discharge of 20% of capacity.

Table 8-1. Voltage Divider - Choosing Resistor Values for Low Power Consumption

Resistor Divider Values		Current Consumption	Battery Lifetime for a Given Battery Capacity	
R1	**R2**	V_{BAT} = 7.4V	1650 mAh	2950 mAh
8.66 kΩ	1 kΩ	0.76 mA	~71 days	~128 days
8.66 MΩ	1 MΩ	0.76 µA	~198 years	~351 years

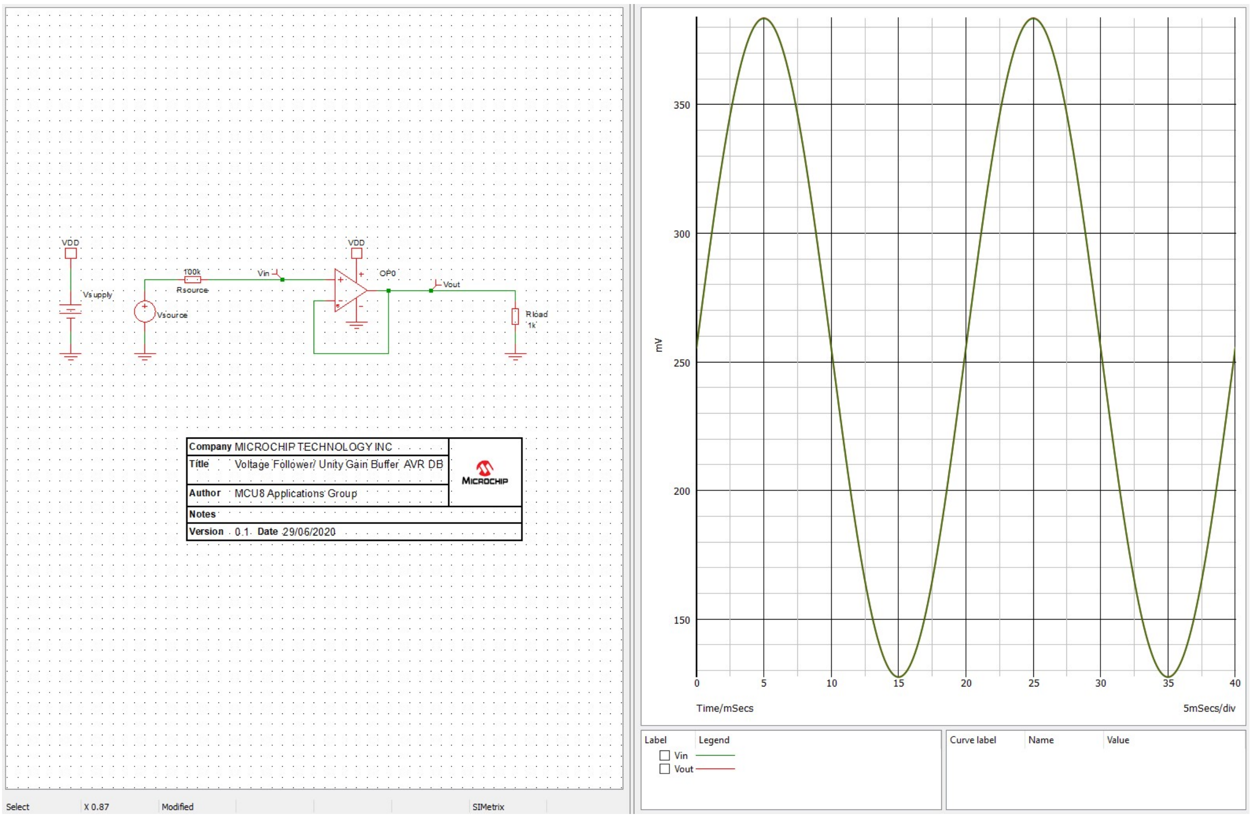
8.2 Voltage Follower Analog Simulation with MPLAB® Mindi™

In MPLAB® Mindi™, open **Voltage_Follower.wxsch** schematic from the GitHub repository.

View Code Examples on GitHub
Click to browse repositories

The circuit shown in the left half of Figure 8-3 is driven by a 50 Hz, 256 mV_{PP} sinusoidal signal source with 256 mV DC offset. The op amp is powered by a 5V battery source.

Figure 8-3. Voltage Follower - Analog Simulation with MPLAB® Mindi™



Simulate by selecting; **Simulator → Run Schematic** or **press Function Key F9**. The resulting simulation output graph is shown in the right half of Figure 8-3. It can be noticed that the output signal follows closely the input signal. It is precisely what would be expected from a voltage follower application.

8.3 Register Configuration

The previous section briefly explored an MPLAB® Mindi™ simulation of a voltage follower circuit. In this section, the relevant AVR DB OPAMP registers are configured to enable the voltage follower configuration and operation. Two options are considered:

- Positive input of the op amp is connected to the device's input pin
- Positive input of the op amp is internally connected to the output of the digital-to-analog converter (DAC). This configuration is used in the accompanying code example.

Table 8-2. Voltage Follower with Positive Input Connected to Device's Pin

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	INP	OUT	OFF	WIP0	OFF

Table 8-3. Voltage Follower with Positive Input Connected to Internal DAC

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	DAC	OUT	OFF	WIP0	OFF

In the case of the Voltage follower, the resistor ladder multiplexer register will remain configured to `0x00`.

Figure 8-4. OPAMP.OPnRESMUX - Configure Resistor Ladder Multiplexer

Bit	7	6	5	4	3	2	1	0
	MUXWIP[2:0]			MUXBOT[2:0]			MUXTOP[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:5 – MUXWIP[2:0] Multiplexer for Wiper
This bit field selects the resistor ladder wiper (potentiometer) position.

Value	Name	Description
0x0	WIP0	R1 = 15R, R2 = 1R
0x1	WIP1	R1 = 14R, R2 = 2R
0x2	WIP2	R1 = 12R, R2 = 4R
0x3	WIP3	R1 = 8R, R2 = 8R
0x4	WIP4	R1 = 6R, R2 = 10R
0x5	WIP5	R1 = 4R, R2 = 12R
0x6	WIP6	R1 = 2R, R2 = 14R
0x7	WIP7	R1 = 1R, R2 = 15R

Bits 4:2 – MUXBOT[2:0] Multiplexer for Bottom
This bit field selects the analog signal connected to the bottom resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	INP	Positive input pin for OPn
0x2	INN	Negative input pin for OPn
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x4	LINKOUT	OP[n-1] output (Setting only available for OP1)(1)
0x5	GND	Ground
Other	-	Reserved

Note: When selecting LINKOUT for OP0, MUXBOT is connected to the output of OP2.

Bits 1:0 – MUXTOP[1:0] Multiplexer for Top
This bit field selects the analog signal connected to the top resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	OUT	OPn output
0x2	VDD	V_{DD}
Other	-	Reserved

```
OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;
```

In the voltage follower code example accompanying this document, the output of the op amp is connected to the negative input. The positive input of the op amp is internally connected to the DAC output.

Figure 8-5. OPAMP.OPnINMUX - Configure the Input Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXNEG[2:0]				MUXPOS[2:0]		
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:4 – MUXNEG[2:0] Multiplexer for Negative Input
This bit field selects which analog signal is connected to the inverting (-) input of OPn.

Value	Name	Description
0x0	INN	Negative input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	OUT	OPn output (unity gain)
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
Other	-	Reserved

Bits 2:0 – MUXPOS[2:0] Multiplexer for Positive Input
This bit field selects which analog signal is connected to the non-inverting (+) input of OPn.

Value	Name	Description
0x0	INP	Positive input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x3	GND	Ground
0x4	VDDDIV2	$V_{DD/2}$
0x5	LINKOUT	OP[n-1] output (Setting only available for OP1 and OP2)
0x6	LINKWIP	Wiper from OP0's resistor ladder (Setting only available for OP2)
Other	-	Reserved

```
OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_DAC_gc;
```

The voltage follower initialization code will look as follows:

```
void OPAMP0_init (void)
{
    /* Configure the Timebase */
    OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;

    /* Configure the voltage input range */
    OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;

    /* Configure the Op Amp n Control A */
    OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;

    /* Configure the Op Amp n Input Multiplexer */
    OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_DAC_gc;

    /* Configure the Op Amp n Resistor Wiper Multiplexer */
    OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;

    /* Configure the Op Amp n Settle Time*/
    OPAMP.OP0SETTLE = 0x7F;

    /* Enable OPAMP peripheral */
    OPAMP.CTRLA = OPAMP_ENABLE_bm;

    /* Wait for the operational amplifiers to settle */
    while (OPAMP.OP0STATUS & OPAMP_SETTLED_bm)
    {
        ;
    }
}
```

The code for this example is available in the **voltage-follower** folder in these github repositories

View Code Example on GitHub
Click to browse repository

8.4 Data Streaming to MPLAB® *Data Visualizer*

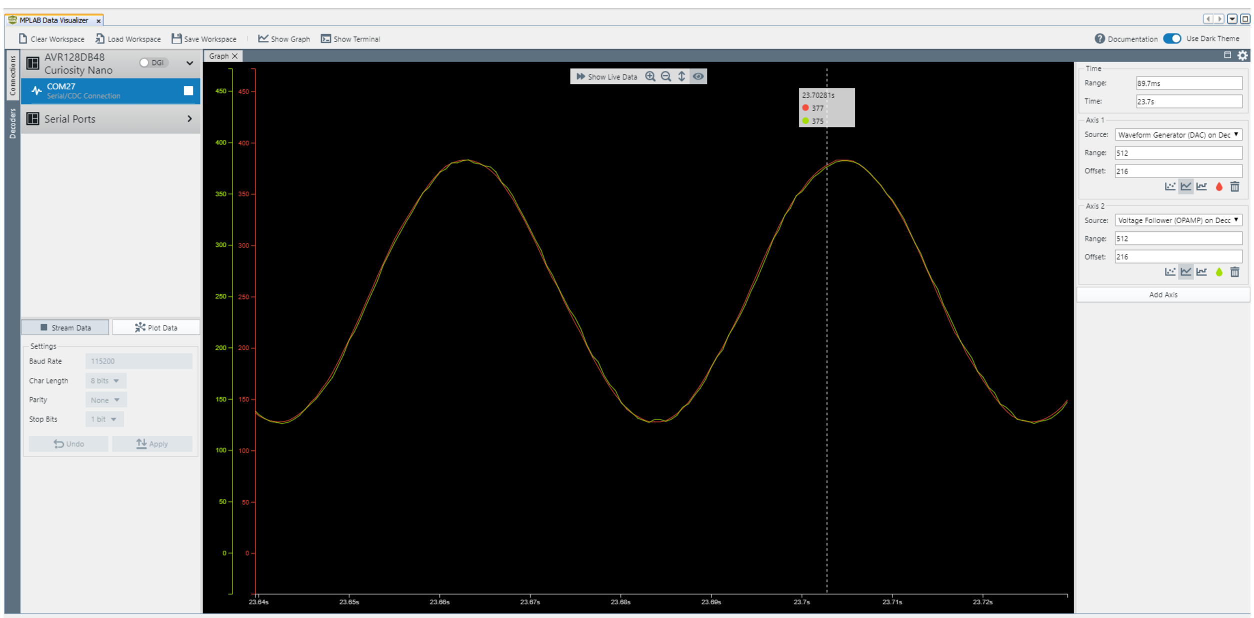
In this section, the AVR DB on the Curiosity Nano will demonstrate such operation, using one of its internal op amps. Hence the microcontroller is programmed with an application that:

- Configures OPAMP0 as a voltage follower
- Uses the on-chip DAC and TCB0 to generate a 50 Hz sine wave, which is fed into the positive input of OPAMP0
- Employs the on-chip TCB1 and ADC for sampling the output of the OPAMP0
- Data streams the ADC result and the DAC sample, via USART, to MPLAB *Data Visualizer*

There are no hardware requirements for this application as all above-mentioned modules and connections are internal to the AVR DB.

To get the demo up and running, follow the steps outlined in 6. MPLAB® Data Visualizer. The workspace-file to load is *voltage_follower.json*, available in the GitHub repository. Figure 8-6 shows an example of the expected result.

Figure 8-6. Voltage Follower - Data Visualizer Output Graph

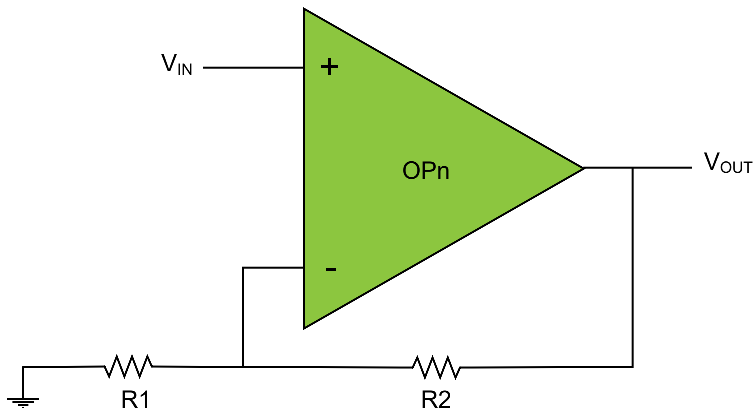


The graph shows, in red, the DAC waveform, while in green, the sampled output of the OPAMP0 configured as a voltage follower. As expected from the simulation step, the two signals are overlapping in magnitude and time. The differences in signal magnitude, between the DAC generated waveform and the ADC sampled one, are due to the accumulated errors relating to the DAC, ADC and op amp (such discussion is beyond the scope of this document).

9. Non-Inverting Programmable Gain Amplifier

Figure 9-1 displays an op amp in a non-inverting amplifier configuration. The main difference from the previous voltage follower configuration is the presence of the R2/R1 resistor divider in the negative feedback loop. The resistor divider allows for the upward scaling of the input signal. This adjustment can be done by programming the R2/R1 ratio, hence the name of Programmable Gain Amplifier (PGA).

Figure 9-1. Non-Inverting Programmable Gain Amplifier



The output voltage is given by the formula:

$$V_{OUT} = \left(1 + \frac{R_2}{R_1}\right) V_{IN}$$

The gain (amplification) of the configuration is given by the formula:

$$Gain = 1 + \frac{R_2}{R_1}$$

As in the case of the voltage follower, the non-inverting programmable gain amplifier is applicable for coupling a high input impedance with a low output impedance. It is perfectly suited as an intermediary stage between a low voltage signal coupled to an ADC.

9.1 Use Case

The non-inverting amplifier (as mentioned in the previous section) has the merit of being able to:

- Couple a high impedance signal source to a low impedance signal sink
- Take an input signal and amplify it by a certain amount, *G (gain)*. The gain is directly proportional to the R2/R1 resistor value ratio present in the feedback loop (see Figure 9-1).

These two properties make the non-inverting amplifier an ideal choice as a pre-input stage to an ADC because:

- The input impedance to an ADC has a low value hence the non-inverting amplifier acts as a buffer
- Increases the dynamic range of the input signal improving the sampling resolution

Consider the case when the input signal to an ADC is 25 mV referred to the system's ground. The ADC has a 10-bit resolution, and the voltage reference has been selected as 1.024V. It means, as per the data sheet, that the quantization error, of one least significant bit (LSb), translates to 1 mV, which represents 4% of the input signal. If an accurate measurement is desired, this can represent a high source of error. However, it can be improved by amplifying the input signal before sampling it via the ADC. With a gain of 16, the input signal will be amplified to 400 mV before sampling. It follows that the quantization error of 1 mV, represents only 0.25% of the input signal, which is a considerable improvement.

It is common for applications requiring a non-inverting amplifier to have a fixed gain, set via the external resistors, as part of the negative feedback loop. These applications assume that the input signal will fit within a fixed, predefined range. However, such a setup is limiting if the input signal has a high dynamic range, or the transducers (sensors) exhibit a change in their output signal over time, due to external (e.g., environmental) or internal (e.g., aging) factors. For such cases, it is beneficial to be able to modify the gain of the amplifier without a change in components. A change in gain either upwards or downwards will bring the signal of interest within its specified range once more. This can be done through a programmable gain amplifier (PGA) where the feedback resistor ratio can be adjusted between several values. It is likely that these applications involve a control loop where the algorithm is constantly monitoring the control and feedback signals and is able to make a decision when a different gain is required. PGAs have a wide area of applications such as audio and voice, data acquisition, industrial and medical instrumentation, lighting, motor control, power control, and test equipment.

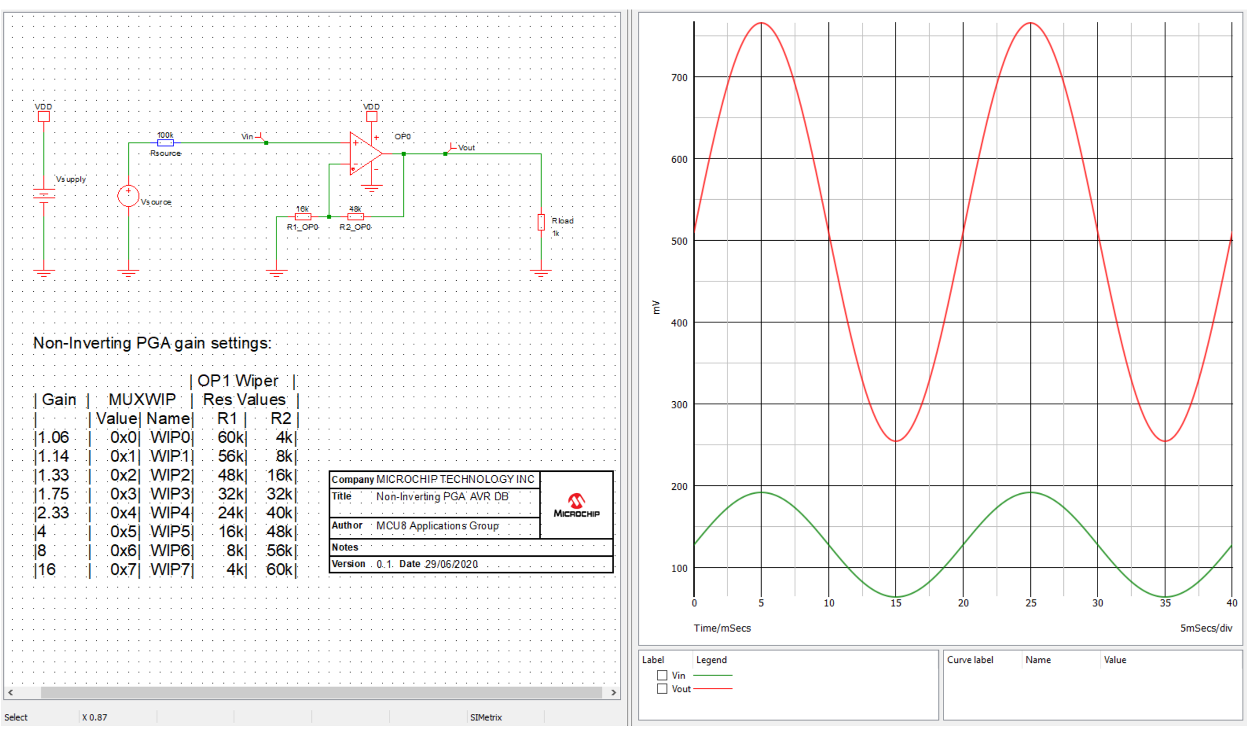
9.2 Non-Inverting PGA Analog Simulation with MPLAB® Mindi™

In MPLAB® Mindi™, open **Non_Inverting_PGA(AVR_DB).wxsch** schematic from the GitHub repository.

View Code Examples on GitHub
Click to browse repositories

The circuit shown in the left half of Figure 9-2 is driven by a 50 Hz, 128 mV_{PP} sinusoidal signal source with 128 mV DC offset. The op amp is powered by a 3.3V battery source.

Figure 9-2. Non-Inverting PGA - Analog Simulation with MPLAB® Mindi™



Simulate by selecting; **Simulator** → **Run Schematic** or **press Function Key F9**. The resulting simulation output graph is shown in the right half of Figure 9-2. It can be noticed that the output signal (in red) is twice the input signal (in green) while keeping the signal in phase. It is precisely what would be expected from a gain amplifier configuration.

9.3 Register Configuration

The previous section briefly explored an MPLAB® Mindi™ simulation of a non-inverting programmable gain amplifier circuit. In this section, the relevant AVR DB OPAMP registers are configured to enable the non-inverting PGA configuration and operation as follows:

- Positive input of the op amp is connected to the output of the internal digital to analog converter (DAC)
- Negative input of the op amp is connected to the wiper position of the internal resistor ladder
- Bottom part of the resistor ladder is connected to ground. The wiper position determines the gain, and the top part of the resistor ladder is connected to the output of the op amp.

Table 9-1 summarizes the necessary settings to set one of the internal op amps in a non-inverting PGA configuration.

Table 9-1. Non-Inverting PGA with Positive Input Connected to Internal DAC

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	DAC	WIP	GND	Setting determines gain	OUT

In the case of the non-inverting PGA, the resistor ladder multiplexer register will be configured to `0x75`.

Figure 9-3. OPAMP.OPnRESMUX - Configure Resistor Ladder Multiplexer

Bit	7	6	5	4	3	2	1	0
	MUXWIP[2:0]			MUXBOT[2:0]			MUXTOP[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:5 – MUXWIP[2:0] Multiplexer for Wiper
This bit field selects the resistor ladder wiper (potentiometer) position.

Value	Name	Description
0x0	WIP0	R1 = 15R, R2 = 1R
0x1	WIP1	R1 = 14R, R2 = 2R
0x2	WIP2	R1 = 12R, R2 = 4R
0x3	WIP3	R1 = 8R, R2 = 8R
0x4	WIP4	R1 = 6R, R2 = 10R
0x5	WIP5	R1 = 4R, R2 = 12R
0x6	WIP6	R1 = 2R, R2 = 14R
0x7	WIP7	R1 = 1R, R2 = 15R

Bits 4:2 – MUXBOT[2:0] Multiplexer for Bottom
This bit field selects the analog signal connected to the bottom resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	INP	Positive input pin for OPn
0x2	INN	Negative input pin for OPn
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x4	LINKOUT	OP[n-1] output (Setting only available for OP1)(1)
0x5	GND	Ground
Other	-	Reserved

Note: When selecting LINKOUT for OP0, MUXBOT is connected to the output of OP2.

Bits 1:0 – MUXTOP[1:0] Multiplexer for Top
This bit field selects the analog signal connected to the top resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	OUT	OPn output
0x2	VDD	V_{DD}
Other	-	Reserved

```
OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_GND_gc | OPAMP_OP0RESMUX_MUXWIP_WIP3_gc |
OPAMP_OP0RESMUX_MUXTOP_OUT_gc;
```

In the non-inverting PGA code example accompanying this document, the output of the operational amplifier is connected to the negative input via the resistor ladder. The positive input of the operational amplifier is internally connected to the DAC output.

Figure 9-4. OPAMP.OPnINMUX - Configure the Input Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXNEG[2:0]				MUXPOS[2:0]		
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:4 – MUXNEG[2:0] Multiplexer for Negative Input
This bit field selects which analog signal is connected to the inverting (-) input of OPn.

Value	Name	Description
0x0	INN	Negative input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	OUT	OPn output (unity gain)
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
Other	-	Reserved

Bits 2:0 – MUXPOS[2:0] Multiplexer for Positive Input
This bit field selects which analog signal is connected to the non-inverting (+) input of OPn.

Value	Name	Description
0x0	INP	Positive input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x3	GND	Ground
0x4	VDDDIV2	$V_{DD/2}$
0x5	LINKOUT	OP[n-1] output (Setting only available for OP1 and OP2)
0x6	LINKWIP	Wiper from OP0's resistor ladder (Setting only available for OP2)
Other	-	Reserved

```
OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_WIP_gc | OPAMP_OP0INMUX_MUXPOS_DAC_gc;
```

The non-inverting PGA initialization code will look as follows:

```
void OPAMP0_init (void)
{
    /* Configure the Timebase */
    OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;

    /* Configure the voltage input range */
    OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;

    /* Configure the Op Amp n Control A */
    OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;

    /* Configure the Op Amp n Input Multiplexer */
    OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_WIP_gc | OPAMP_OP0INMUX_MUXPOS_DAC_gc;

    /* Configure the Op Amp n Resistor Wiper Multiplexer */
    /* WIP3 => R1 = 8R, R2 = 8R */
    /* Gain = 1 + R2/R1 = 2 */
    OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_GND_gc | OPAMP_OP0RESMUX_MUXWIP_WIP3_gc |
OPAMP_OP0RESMUX_MUXTOP_OUT_gc;

    /* Configure the Op Amp n Settle Time*/
    OPAMP.OP0SETTLE = 0x7F;

    /* Enable OPAMP peripheral */
    OPAMP.CTRLA = OPAMP_ENABLE_bm;

    /* Wait for the operational amplifiers to settle */
    while (OPAMP.OP0STATUS & OPAMP_SETTLED_bm)
    {
        ;
    }
}
```

The code for this example is available in the **non-inverting-pga** folder in these github repositories

View Code Example on GitHub
Click to browse repository

9.4 Data Streaming to MPLAB® *Data Visualizer*

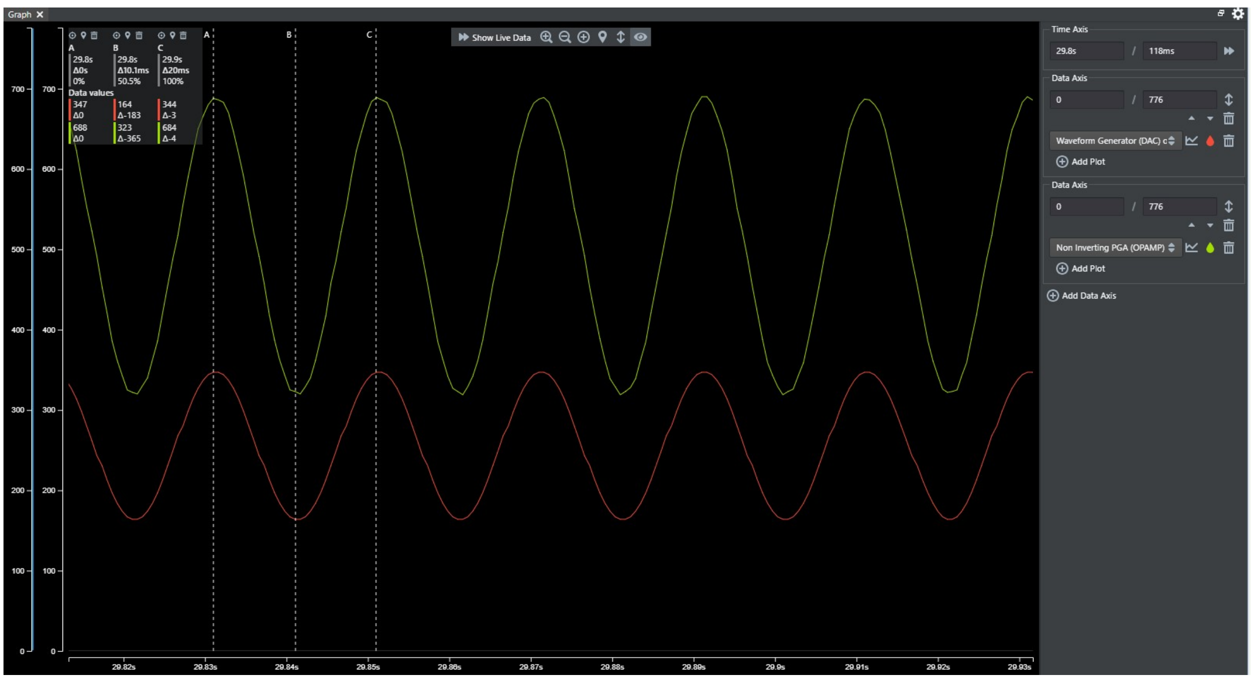
In this section, the AVR DB on the Curiosity Nano will demonstrate non-inverting PGA operation, using one of its internal op amps. Hence the microcontroller is programmed with an application that:

- Configures OPAMP0 as a non-inverting gain amplifier
- Uses the on-chip DAC and TCB0 to generate a 50 Hz sine wave, which is fed into the positive input of OPAMP0
- Employs the on-chip TCB1 and ADC for sampling the output of the OPAMP0
- Data streams the ADC result and the DAC sample, via USART, to MPLAB *Data Visualizer*

There are no hardware requirements for this application, as all above-mentioned modules and connections are internal to the AVR DB.

To get the demo up and running, follow the steps outlined in 6. MPLAB® Data Visualizer. The workspace-file to load is *non_inverting_PGA.json*, available in the GitHub repository. Figure 9-5 shows an example of the expected result.

Figure 9-5. Non-Inverting PGA - Data Visualizer Output Graph

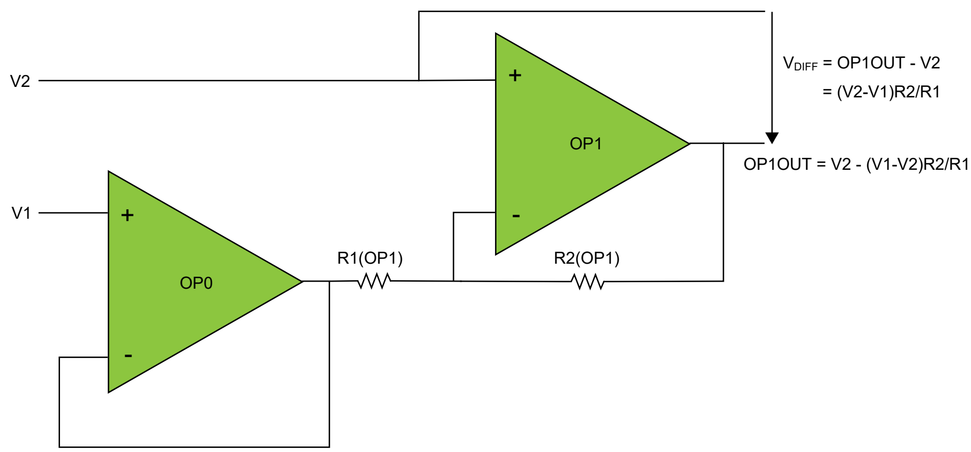


The graph shows, in red, the DAC waveform while, in lime green, the sampled output of the OPAMP0, configured as non-inverting PGA. As expected, from the simulation step, the output signal of the amplifier has been amplified by a factor of 2, while keeping the input and output signal in phase.

10. Differential Amplifier

Figure 10-1 displays two op amps connected in an differential amplifier configuration. As seen in the figure the differential amplifier has two input signals V1 and V2 and two output signals OP1OUT and V2. The difference between the two output signals is proportional to the difference between the input signals. The ratio or the gain between the two differential signals is set by the ratio between R2 and R1. This way the differential amplifier can amplify a differential signal.

Figure 10-1. Differential Amplifier



The output voltage OP1OUT is given by the formula:

$$V_{OP1OUT} = V_2 - \left(V_1 - V_2\right)\frac{R_2}{R_1}$$

The differential signal is given by the formula:

$$V_{DIFF} = OP1OUT - V_2 = \left(V_2 - V_1\right)\frac{R_2}{R_1}$$

The gain is:

$$Gain = \frac{R_2}{R_1}$$

The differential amplifier is suitable for amplifying a small differential signal before it is converted to a digital signal by a differential ADC.

10.1 Use Case

Using a shunt resistor for measuring a circuit's current consumption is a common application for microcontrollers with a differential ADC. The ADC can measure the voltage drop over the resistor, and using ohm's law, the current can be calculated from the measured voltage and the known resistance of the shunt resistor:

$$I_m = \frac{V_m}{R_s}$$

Introducing a series resistor into the circuit has two adverse effects:

- Increases the power consumption of the circuit
- A voltage drop over the added resistor

Both effects can be mitigated by choosing a resistor R_S with as small as possible resistance, especially important when considering low power applications. The downside of choosing a low resistance is that measuring the voltage drop becomes more difficult as a change in current results in a very small change in voltage. Such an issue can be alleviated by using a differential amplifier to amplify the voltage drop across the current sense resistor and thus increasing the resolution of the measurement. The effects of the resistor's size on the measurement's resolution and the power consumption can be seen in Table 10-1.

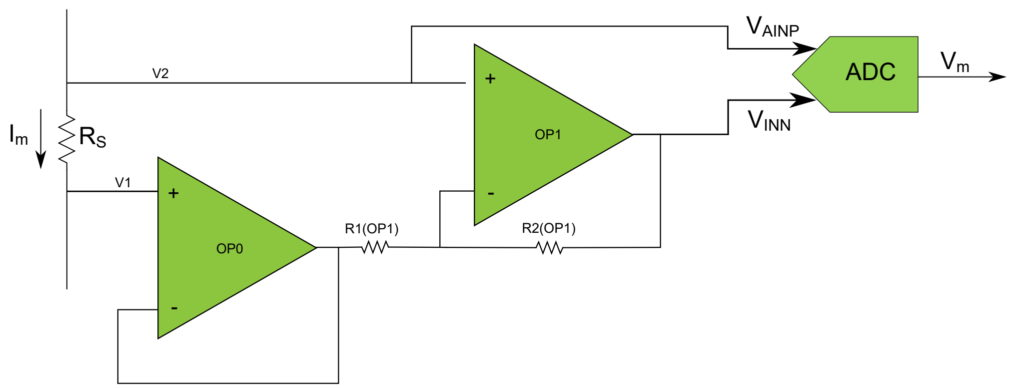
Table 10-1. Relation Between Resistor Size and Power Consumption and Resolution

Resistor$_S$ [Ω]	Voltage Drop [mV]	Power Consumtion [mW]	Resolution no Amplification [mA/bit]	Resolution 7x Amplification [mA/bit]
10	100	1000	50	7.14
1	10	100	500	71.4
0.1	1	10	5000	714

Note: The table assumes a current of 10 mA and a 12-bit differential ADC with a reference voltage of 1.024V.

Figure 10-2 shows how a differential amplifier can be placed in between the shunt resistor and the ADC to measure the amplified signal.

Figure 10-2. Differential Amplifier Used in a Shunt Current Measurement



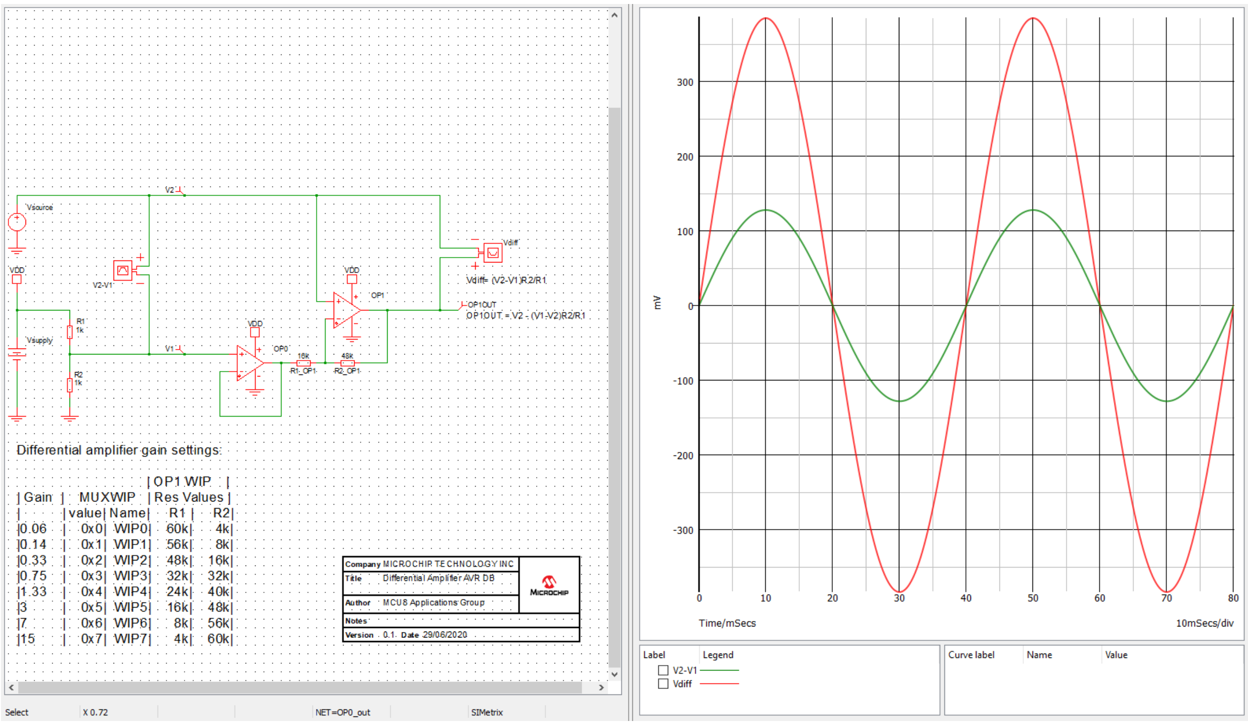
10.2 Differential Amplifier Analog Simulation with MPLAB® Mindi™

In MPLAB® Mindi™, open **Differential_Amplifier.wxsch** schematic from the GitHub repository.

View Code Examples on GitHub
Click to browse repositories

The circuit shown in the left half of Figure 10-3, is driven by $V_{DD}/2$ for the first op amp and a 25 Hz 256 mV_{PP} sinusoidal signal source with 1.65V DC offset for the second op amp. Both op amps are powered by a 3.3V power supply.

Figure 10-3. Differential Amplifier - Analog Simulation with MPLAB® Mindi™



Simulate by selecting; **Simulator** → **Run Schematic** or **press Function Key F9**. The resulting simulation output graph is shown in the right half of Figure 10-3. It can be seen that the peaks of the bottom signal representing the differential output are ±384 mV, which are as expected as the peaks of the input sinusoidal are ±128 mV relative to $V_{DD}/2$.

10.3 Register Configuration

The previous section briefly explored an MPLAB® Mindi™ simulation of a differential amplifier circuit. In this section, the relevant AVR DB OPAMP registers are configured to enable the differential amplifier configuration and operation. Two options are considered:

- Positive inputs of the op amps are connected to the device's input pins
- The positive input of op amp n is connected to V_{DD} divided by two, and op amp n+1 is connected to the DAC[(1)]. This configuration is used in the accompanying code example.

Table 10-2. Differential Amplifier with Positive Inputs Connected to Device's Pins

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	INP	OUT	OFF	WIP0	OFF
OPn+1[(1)]	INP	WIP	LINKOUT	Setting determines gain	OUT

Table 10-3. Differential Amplifier with Positive Inputs Connected to V_{DD} Divided by Two and Internal DAC

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OPn	VDDDIV2	OUT	OFF	WIP0	OFF
OPn+1[(1)]	DAC	WIP	LINOUT	Setting determines gain	OUT

Note:
1. If n = 2 then OPn+1 is OP0.

In the case of the differential amplifier, the resistor ladder multiplexer registers will be configured to `0x00` and `0xB1` for OP0 and OP1, respectively. This gives a gain of 3.

Figure 10-4. OPAMP.OPnRESMUX - Configure Resistor Ladder Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXWIP[2:0]			MUXBOT[2:0]		MUXTOP[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:5 – MUXWIP[2:0] Multiplexer for Wiper
This bit field selects the resistor ladder wiper (potentiometer) position.

Value	Name	Description
0x0	WIP0	R1 = 15R, R2 = 1R
0x1	WIP1	R1 = 14R, R2 = 2R
0x2	WIP2	R1 = 12R, R2 = 4R
0x3	WIP3	R1 = 8R, R2 = 8R
0x4	WIP4	R1 = 6R, R2 = 10R
0x5	WIP5	R1 = 4R, R2 = 12R
0x6	WIP6	R1 = 2R, R2 = 14R
0x7	WIP7	R1 = 1R, R2 = 15R

Bits 4:2 – MUXBOT[2:0] Multiplexer for Bottom
This bit field selects the analog signal connected to the bottom resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	INP	Positive input pin for OPn
0x2	INN	Negative input pin for OPn
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x4	LINKOUT	OP[n-1] output (Setting only available for OP1)[(1)]
0x5	GND	Ground
Other	-	Reserved

Note: When selecting LINKOUT for OP0, MUXBOT is connected to the output of OP2.

Bits 1:0 – MUXTOP[1:0] Multiplexer for Top
This bit field selects the analog signal connected to the top resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	OUT	OPn output
0x2	VDD	V_{DD}
Other	-	Reserved

```
OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;
OPAMP.OP1RESMUX = OPAMP_OP1RESMUX_MUXBOT_LINKOUT_gc | OPAMP_OP1RESMUX_MUXWIP_WIP5_gc |
OPAMP_OP1RESMUX_MUXTOP_OUT_gc;
```

In the differential amplifier code example accompanying this document, the positive input of OP0 is connected to V_{DD} divided by two, and OP1 is connected to the DAC.

Figure 10-5. OPAMP.OPnINMUX - Configure the Input Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXNEG[2:0]				MUXPOS[2:0]		
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:4 – MUXNEG[2:0] Multiplexer for Negative Input
This bit field selects which analog signal is connected to the inverting (-) input of OPn.

Value	Name	Description
0x0	INN	Negative input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	OUT	OPn output (unity gain)
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
Other	-	Reserved

Bits 2:0 – MUXPOS[2:0] Multiplexer for Positive Input
This bit field selects which analog signal is connected to the non-inverting (+) input of OPn.

Value	Name	Description
0x0	INP	Positive input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x3	GND	Ground
0x4	VDDDIV2	$V_{DD/2}$
0x5	LINKOUT	OP[n-1] output (Setting only available for OP1 and OP2)
0x6	LINKWIP	Wiper from OP0's resistor ladder (Setting only available for OP2)
Other	-	Reserved

```
OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_VDDDIV2_gc;
OPAMP.OP1INMUX = OPAMP_OP1INMUX_MUXNEG_WIP_gc | OPAMP_OP1INMUX_MUXPOS_DAC_gc;
```

The voltage follower initialization code will look as follows:

```
void OPAMP_init ()
{
    /* Configure the Timebase */
    OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;

    /* Configure the voltage input range */
    OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;

    /* Configure the OP0 */
    OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
    OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_VDDDIV2_gc;
    OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_OFF_gc | OPAMP_OP0RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP0RESMUX_MUXTOP_OFF_gc;
    OPAMP.OP0SETTLE = 0x7F;

    /* Configure the OP1*/
    OPAMP.OP1CTRLA = OPAMP_OP1CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
    OPAMP.OP1INMUX = OPAMP_OP1INMUX_MUXNEG_WIP_gc | OPAMP_OP1INMUX_MUXPOS_DAC_gc;
    OPAMP.OP1RESMUX = OPAMP_OP1RESMUX_MUXBOT_LINKOUT_gc | OPAMP_OP1RESMUX_MUXWIP_WIP5_gc |
OPAMP_OP1RESMUX_MUXTOP_OUT_gc;
    OPAMP.OP1SETTLE = 0x7F;

    /* Enable OPAMP peripheral */
    OPAMP.CTRLA = OPAMP_ENABLE_bm;

    /* Wait for the operational amplifiers to settle */
    while ((OPAMP.OP0STATUS & OPAMP_SETTLED_bm) & (OPAMP.OP1STATUS & OPAMP_SETTLED_bm))
    {
        ;
    }
}
```

The code for this example is available in the **differential-amplifier** folder in these github repositories

View Code Example on GitHub
Click to browse repository

10.4 Data Streaming to MPLAB® *Data Visualizer*

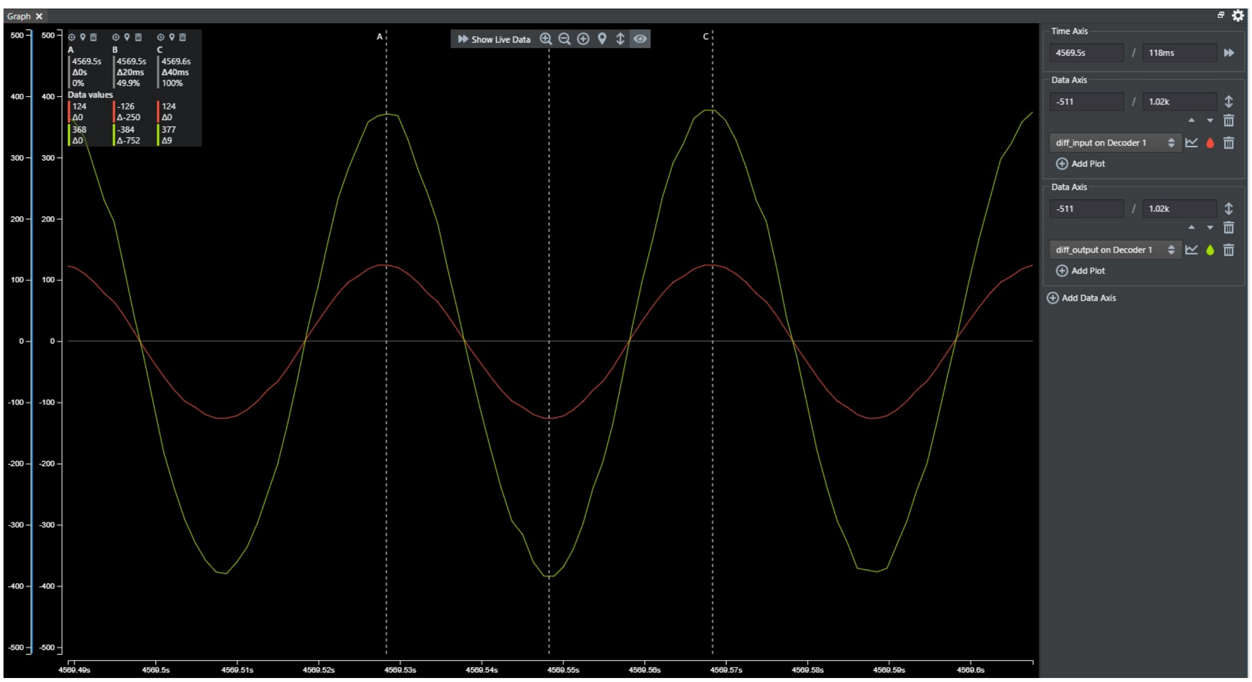
In this section, the AVR DB on the Curiosity Nano will demonstrate differential amplifier operation, using two of its internal op amps. Hence the microcontroller is programmed with an application that:

- Configures OPAMP0 and OPAMP1 to create a differential amplifier configuration
- The positive input of OPAMP0 is connected to $V_{DD}/2$
- Uses the on-chip DAC and TCB0 to generate a 25 Hz sine wave, which is fed into the positive input of OPAMP1
- Employs the on-chip TCB1 and ADC for sampling the differential signal between the DAC and the output from OPAMP1
- Data streams the ADC result, and the DAC sample subtracted from V_{DD} /2, via USART, to MPLAB *Data Visualizer*

There are no hardware requirements for this application as all the above mentioned modules, and connections are internal to the AVR DB.

To get the demo up and running, follow the steps outlined in 6. MPLAB® Data Visualizer. The workspace-file to load is *differential_amplifier.json*, available in the GitHub repository. Figure 10-6 shows an example of the expected result.

Figure 10-6. Differential Amplifier - Data Visualizer Output Graph

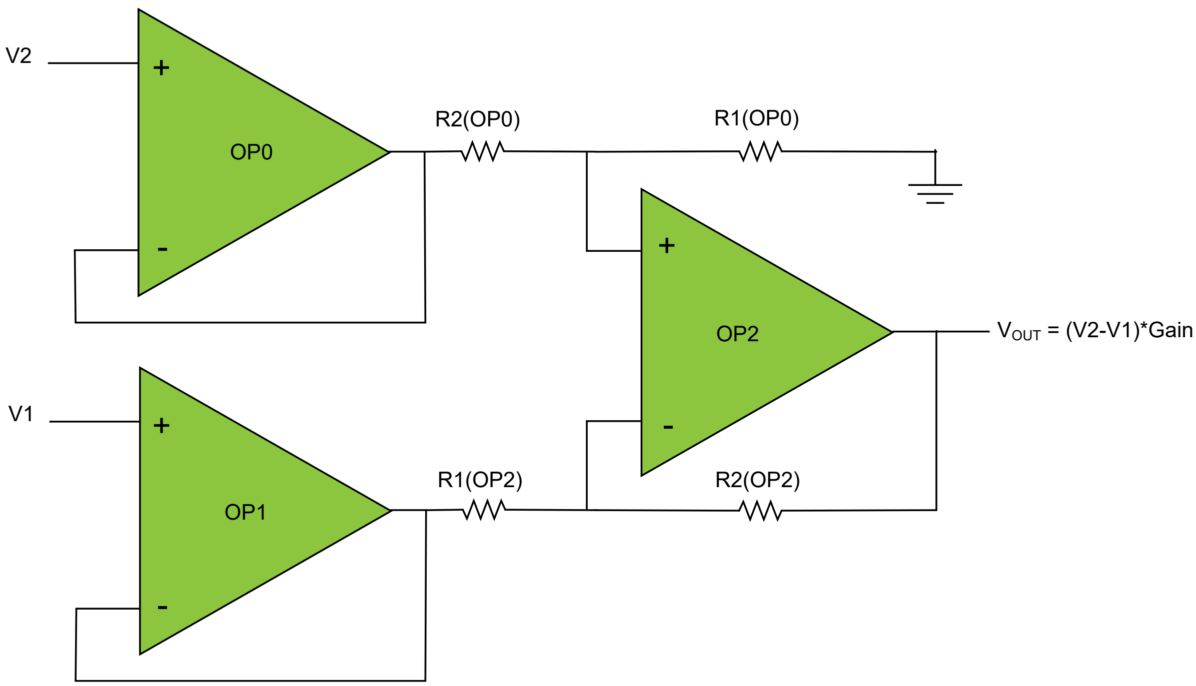


The graph in Figure 10-6 shows in red the DAC waveform relative to $V_{DD}/2$, while the lime green is the sampled output of the OPAMP1 relative to the DAC output. As expected, from the simulation step, the lime green waveform has an amplitude that is approximately three times larger than the input waveform. The reason the amplitude of the light green waveform is not exactly three times as large as the red one is due to the accumulated errors relating to the DAC, ADC and OPAMP (such discussion is beyond the scope of this document).

11. Instrumentation Amplifier

Figure 11-1 displays the instrumentation amplifier configuration. It has two input signals V1 and V2 and one output signal V_{OUT}. The output signal is directly proportional to the difference between the input signals. OP0 and OP1 op amps act as buffers for the V2-V1, differential input voltage, while OP2 acts as a differential amplifier, a combination of both inverting and non-inverting amplifiers. The gain, represented by the ratio between the output signal and the differential input signal, is given by the ratio between R2 and R1.

Figure 11-1. Instrumentation Amplifier



The output voltage V_{OUT} is given by the formula:

$$V_{OUT} = (V_2 - V_1) \times Gain$$

The gain is given by the wiper position for the resistor ladders for OPAMP0 and OPAMP2 according to Table 11-1.

Table 11-1. Gain Selection for Instrumentation Amplifier

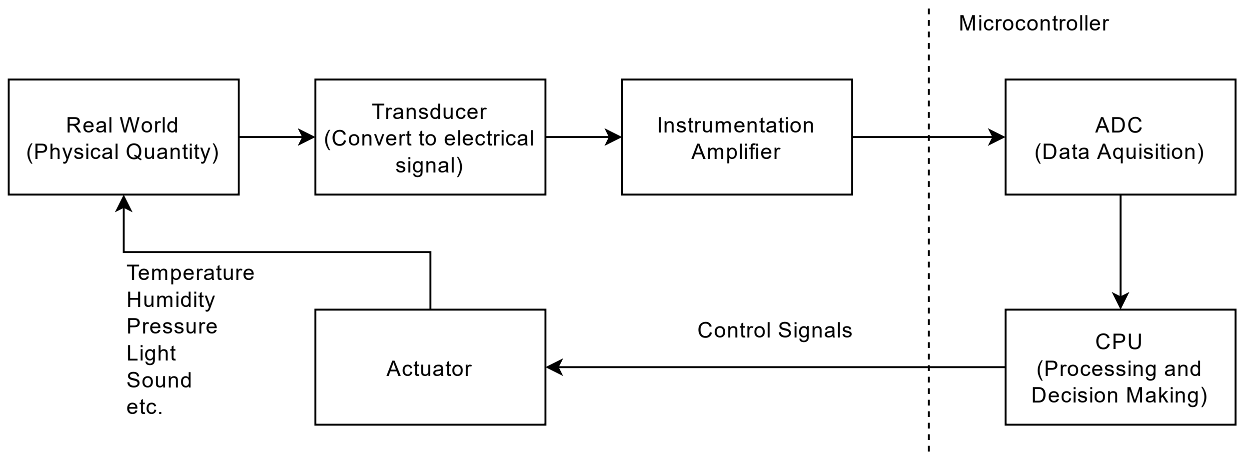
GAIN	OP0RESMUX.MUXWIP	OP2RESMUX.MUXWIP
1/15	0x7	0x0
1/7	0x6	0x1
1/3	0x5	0x2
1	0x3	0x3
3	0x2	0x5
7	0x1	0x6
15	0x0	0x7

As observed, a three op amp instrumentation amplifier takes a differential signal as input and provides a single-ended output signal while ensuring a high input and a low output impedance. Hence it can fit as a signal conditioning element between a differential output sensor and an ADC, which would allow for further processing and decision making.

11.1 Use Case

A multitude of electronic applications requires the measurement and control of physical quantities, such as temperature, humidity, light, air pressure, and so on. The physical quantities are converted into electrical quantities with the help of transducers. The output of a transducer is an electrical signal representing the measured variable. Usually, the transducer's output signal levels are very low and need to be amplified before they are sampled in the data acquisition process. The electric signals are further processed and monitored to offer adequate actions based on their changes. For a general block diagram of a simple measurement and control system see Figure 11-2. The amplifiers used for amplifying electrical signals relating to a physical quantity come under the commonly used term of instrumentation amplifiers. Hence the input to an instrumentation amplifier is the output signal from a transducer.

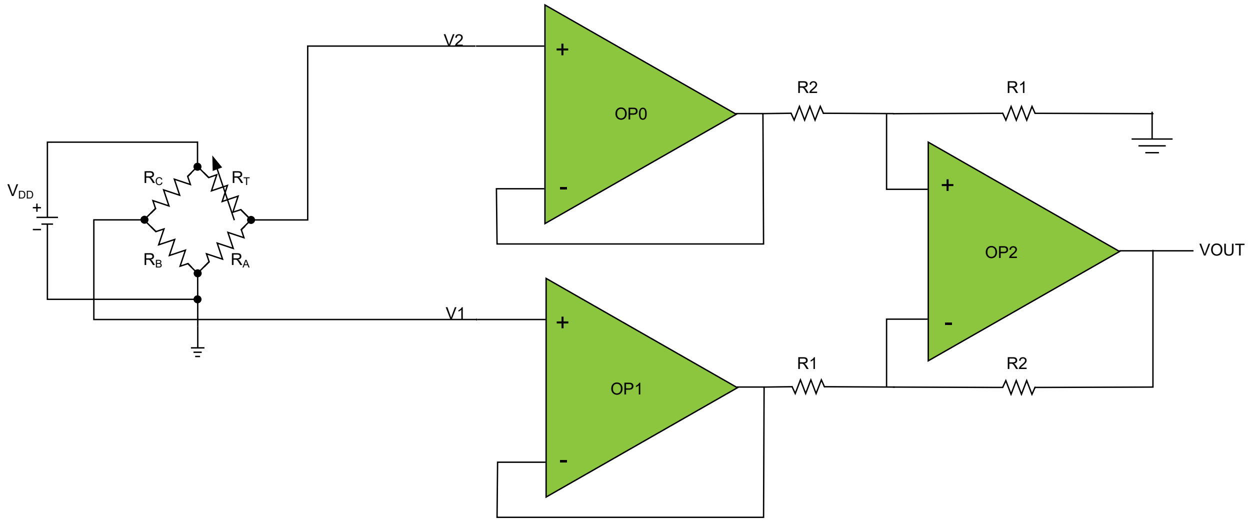
Figure 11-2. Measurement and Control System Signal Flow Block Diagram



Transducers can be classified in different ways. However, an important factor to consider is whether a power source is required or not for their operation. Active transducers do not need an external power source in their operation. They are self-generating the output signal and operate under the energy conversion principle (i.e., photovoltaic, thermoelectric, electromagnetic, piezoelectric, etc.). On the other hand, passive transducers need an external source of energy for their operation. They produce the output signal in the form of variation in an electrical parameter such as resistance, capacitance or inductance.

The Wheatstone bridge (or resistance bridge) circuit can be used to interface various resistive passive transducers to instrumentation amplifiers (Figure 11-3).

Figure 11-3. Wheatstone Bridge and Instrumentation Amplifier



In this circuit, R_A, R_B and R_C are known and given. R_T represents the resistance of the transducer and varies depending on the physical quantity that changes over time. The values for the resistors are chosen for a specific point, which allows for the bridge to be balanced.

When balanced, $\frac{R_C}{R_B} = \frac{R_T}{R_A} = 1$, the output voltage of the Wheatstone bridge (and the differential input to the instrumentation amplifier) is zero, thus the output of the amplifier is zero.

When there is a change in the physical quantity being measured, the voltage V2 will no longer be equal to V1. The resistance of the transducer device changes from R_T to $R_T \pm \Delta R$. This produces a differential voltage input for the instrumentation amplifier, and the output of the amplifier will no longer be zero.

$$V_1 = \frac{R_B}{R_B + R_C} \times V_{DD}$$

$$V_2 = \frac{R_A}{R_A + R_T \pm \Delta R} \times V_{DD}$$

Assuming the resistance are chosen to be of same value, i.e.: $R_A = R_B = R_C = R_T = R$, the differential voltage input becomes:

$$V_2 - V_1 = \frac{R}{2R} \times V_{DD} - \frac{R}{2R \pm \Delta R} \times V_{DD}$$

$$V_2 - V_1 = \frac{\pm \Delta R}{2(2R \pm \Delta R)} \times V_{DD}$$

If the change in resistance ΔR, is much smaller than 2R (ΔR << 2R) the equation can be simplified to:

$$V_2 - V_1 = \frac{\pm \Delta R}{4R} \times V_{DD}$$

It follows that the output from the instrumentation amplifier can be expressed as:

$$V_{OUT} = Gain \times \frac{\pm \Delta R}{4R} \times V_{DD}$$

Which means the instrumentation amplifier's output voltage directly depends on:

- Change in the transducer's resistance, ΔR
- Gain of the amplifier which is given by Table 11-1:

An important aspect, which needs to be considered is that the differential voltage, V_2-V_1, has to be positive at all times. This is a consequence of the op amp rail supply voltage of 0V for the negative rail and +V_{DD} for the positive rail. Hence care must be taken when positioning the transducer in the Wheatstone bridge configuration. The transducer position depends on the negative or positive variation of the resistance when excited.

The Wheatstone bridge and the instrumentation amplifier can be used in a wide variety of sensing applications. Such as:

- Temperature sensor, based on a thermistor
- Force sensor, based on a force-sensitive resistor (FSR)
- Weight scales, based on a strain gauges

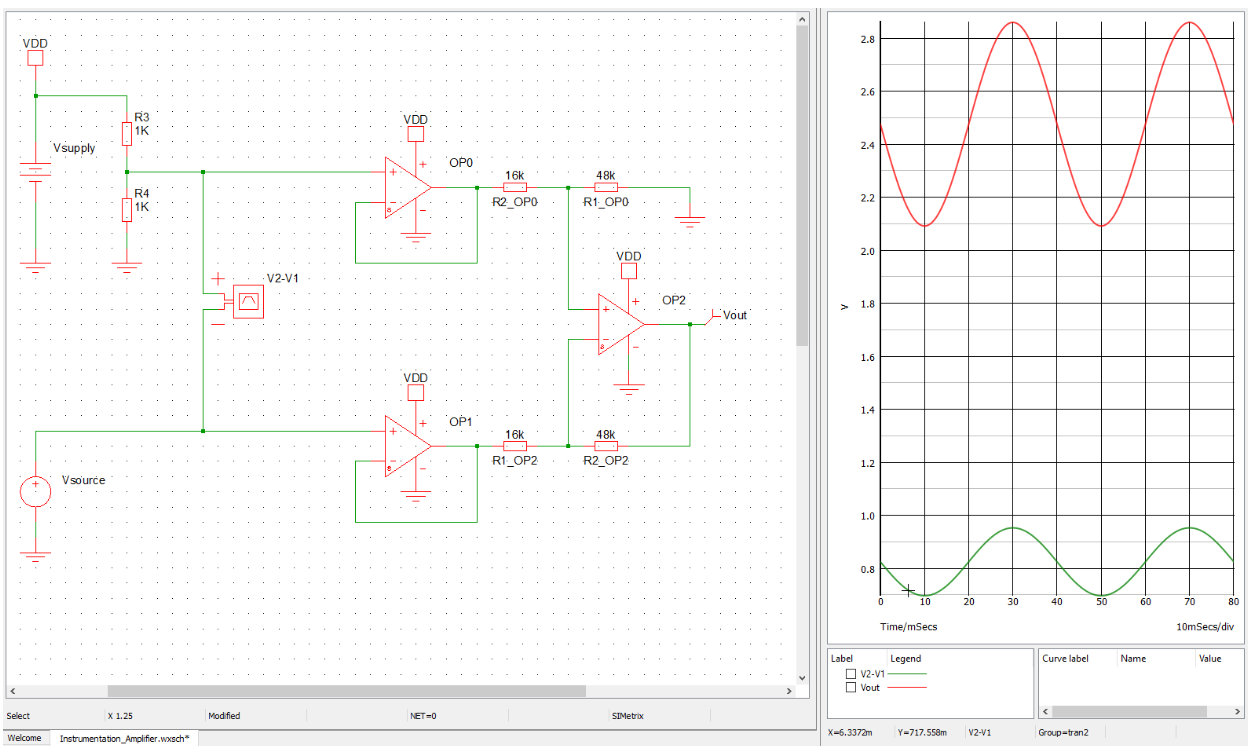
11.2 Instrumentation Amplifier Analog Simulation with MPLAB® Mindi™

In MPLAB® Mindi™, open **Instrumentation_Amplifier.wxsch** schematic from the GitHub repository.

View Code Examples on GitHub
Click to browse repositories

The circuit shown in the left half of Figure 11-4 is driven by a 25 Hz, 256 mV_{PP} sinusoidal differential signal source with 825 mV DC offset. The op amps are powered by a 3.3V voltage source.

Figure 11-4. Instrumentation Amplifier - Analog Simulation with MPLAB® Mindi



Simulate by selecting; **Simulator → Run Schematic** or **press Function Key F9**. The resulting simulation output graph is shown in the right half of Figure 11-4. It can be noticed that the output signal (in red) is three times the input signal (in green), while keeping the signal in phase.

11.3 Register Configuration

The previous section briefly explored an MPLAB® Mindi™ simulation of an instrumentation amplifier circuit. In this section, the relevant AVR DB OPAMP peripheral registers are configured to enable the instrumentation amplifier topology and operation as follows:

- OP0 op amp is set up as a voltage follower, with the positive input is connected internally to $V_{DD/2}$. The top side of the resistor ladder is connected to the output of the op amp, the bottom side to ground, and the wiper position is selected accordingly to the Table 11-1.
- OP1 op amp is constructed as a voltage follower, with the positive input internally connected to the on-board DAC. The resistor ladder is not used.
- OP2 op amp is set up to have as positive input the wiper connection from the OP0 resistor ladder and as negative input the wiper position of its resistor ladder. The topside of the resistor ladder is connected to the output of the op amp, the bottom side to the OP1 op amp and the wiper position is selected accordingly to Table 11-1.

Table 11-2 summarizes the necessary settings to set the three internal op amps in an instrumentation amplifier configuration.

Table 11-2. Instrumentation Amplifier Connected to Internal DAC

	MUXPOS	MUXNEG	MUXBOT	MUXWIP	MUXTOP
OP0	$V_{DD/2}$	OUT	GND	See the table below	OUT
OP1	DAC	OUT	OFF	WIP0	OFF
OP2	LINKWIP (OP0WIP)	WIP	LINKOUT (OP1OUT)	See the table below	OUT

In the case of the instrumentation amplifier, the resistor ladder multiplexer register will be configured separately for each amplifier.

Figure 11-5. OPAMP.OPnRESMUX - Configure Resistor Ladder Multiplexer

Bit	7	6	5	4	3	2	1	0
	MUXWIP[2:0]			MUXBOT[2:0]			MUXTOP[1:0]	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:5 – MUXWIP[2:0] Multiplexer for Wiper
This bit field selects the resistor ladder wiper (potentiometer) position.

Value	Name	Description
0x0	WIP0	R1 = 15R, R2 = 1R
0x1	WIP1	R1 = 14R, R2 = 2R
0x2	WIP2	R1 = 12R, R2 = 4R
0x3	WIP3	R1 = 8R, R2 = 8R
0x4	WIP4	R1 = 6R, R2 = 10R
0x5	WIP5	R1 = 4R, R2 = 12R
0x6	WIP6	R1 = 2R, R2 = 14R
0x7	WIP7	R1 = 1R, R2 = 15R

Bits 4:2 – MUXBOT[2:0] Multiplexer for Bottom
This bit field selects the analog signal connected to the bottom resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	INP	Positive input pin for OPn
0x2	INN	Negative input pin for OPn
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x4	LINKOUT	OP[n-1] output (Setting only available for OP1)(1)
0x5	GND	Ground
Other	-	Reserved

Note: When selecting LINKOUT for OP0, MUXBOT is connected to the output of OP2.

Bits 1:0 – MUXTOP[1:0] Multiplexer for Top
This bit field selects the analog signal connected to the top resistor in the resistor ladder.

Value	Name	Description
0x0	OFF	Multiplexer off
0x1	OUT	OPn output
0x2	VDD	V_{DD}
Other	-	Reserved

```
OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_GND_gc | OPAMP_OP0RESMUX_MUXWIP_WIP2_gc |
OPAMP_OP0RESMUX_MUXTOP_OUT_gc;
OPAMP.OP1RESMUX = OPAMP_OP1RESMUX_MUXBOT_OFF_gc | OPAMP_OP1RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP1RESMUX_MUXTOP_OFF_gc;
OPAMP.OP2RESMUX = OPAMP_OP2RESMUX_MUXBOT_LINKOUT_gc | OPAMP_OP2RESMUX_MUXWIP_WIP5_gc |
OPAMP_OP2RESMUX_MUXTOP_OUT_gc;
```

The same goes for the positive and negative inputs for each amplifier. They need to be separately configured, as outlined in Table 11-2.

Figure 11-6. OPAMP.OPnINMUX - Configure the Input Multiplexer

Bit	7	6	5	4	3	2	1	0
		MUXNEG[2:0]				MUXPOS[2:0]		
Access		R/W	R/W	R/W		R/W	R/W	R/W
Reset		0	0	0		0	0	0

Bits 6:4 – MUXNEG[2:0] Multiplexer for Negative Input
This bit field selects which analog signal is connected to the inverting (-) input of OPn.

Value	Name	Description
0x0	INN	Negative input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	OUT	OPn output (unity gain)
0x3	DAC	DAC output (DAC and DAC output buffer must be enabled)
Other	-	Reserved

Bits 2:0 – MUXPOS[2:0] Multiplexer for Positive Input
This bit field selects which analog signal is connected to the non-inverting (+) input of OPn.

Value	Name	Description
0x0	INP	Positive input pin for OPn
0x1	WIP	Wiper from OPn's resistor ladder
0x2	DAC	DAC output (DAC and DAC output buffer must be enabled)
0x3	GND	Ground
0x4	VDDDIV2	$V_{DD/2}$
0x5	LINKOUT	OP[n-1] output (Setting only available for OP1 and OP2)
0x6	LINKWIP	Wiper from OP0's resistor ladder (Setting only available for OP2)
Other	-	Reserved

```
OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_VDDDIV2_gc;
OPAMP.OP1INMUX = OPAMP_OP1INMUX_MUXNEG_OUT_gc | OPAMP_OP1INMUX_MUXPOS_DAC_gc;
OPAMP.OP2INMUX = OPAMP_OP2INMUX_MUXNEG_WIP_gc | OPAMP_OP2INMUX_MUXPOS_LINKWIP_gc;
```

The instrumentation amplifier initialization code will look as follows:

```
void OPAMP_init (void)
{
    /* Configure the Timebase */
    OPAMP.TIMEBASE = OPAMP_TIMEBASE_US;

    /* Configure the voltage input range */
    OPAMP.PWRCTRL = OPAMP_PWRCTRL_IRSEL_FULL_gc;

    /* Configure the OP0 */
    OPAMP.OP0CTRLA = OPAMP_OP0CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
    OPAMP.OP0INMUX = OPAMP_OP0INMUX_MUXNEG_OUT_gc | OPAMP_OP0INMUX_MUXPOS_VDDDIV2_gc;
    OPAMP.OP0RESMUX =  OPAMP_OP0RESMUX_MUXBOT_GND_gc | OPAMP_OP0RESMUX_MUXWIP_WIP2_gc |
OPAMP_OP0RESMUX_MUXTOP_OUT_gc;
    OPAMP.OP0SETTLE = 0x7F;

    /* Configure the OP1*/
    OPAMP.OP1CTRLA = OPAMP_OP1CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
    OPAMP.OP1INMUX = OPAMP_OP1INMUX_MUXNEG_OUT_gc | OPAMP_OP1INMUX_MUXPOS_DAC_gc;
    OPAMP.OP1RESMUX = OPAMP_OP1RESMUX_MUXBOT_OFF_gc | OPAMP_OP1RESMUX_MUXWIP_WIP0_gc |
OPAMP_OP1RESMUX_MUXTOP_OFF_gc;
    OPAMP.OP1SETTLE = 0x7F;

    /* Configure the OP2*/
    OPAMP.OP2CTRLA = OPAMP_OP2CTRLA_OUTMODE_NORMAL_gc | OPAMP_ALWAYSON_bm;
    OPAMP.OP2INMUX = OPAMP_OP2INMUX_MUXNEG_WIP_gc | OPAMP_OP2INMUX_MUXPOS_LINKWIP_gc;
    OPAMP.OP2RESMUX = OPAMP_OP2RESMUX_MUXBOT_LINKOUT_gc | OPAMP_OP2RESMUX_MUXWIP_WIP5_gc |
OPAMP_OP2RESMUX_MUXTOP_OUT_gc;
    OPAMP.OP2SETTLE = 0x7F;

    /* Enable OPAMP peripheral */
    OPAMP.CTRLA = OPAMP_ENABLE_bm;

    /* Wait for the operational amplifiers to settle */
    while ((OPAMP.OP0STATUS & OPAMP_SETTLED_bm) &&
```



```
                (OPAMP.OP1STATUS & OPAMP_SETTLED_bm) &&
                (OPAMP.OP2STATUS & OPAMP_SETTLED_bm))
        {
            ;
        }
}
```

The code for this example is available in the **instrumentation-amplifier** folder in these github repositories

View Code Example on GitHub
Click to browse repository

11.4 Data Streaming to MPLAB® *Data Visualizer*

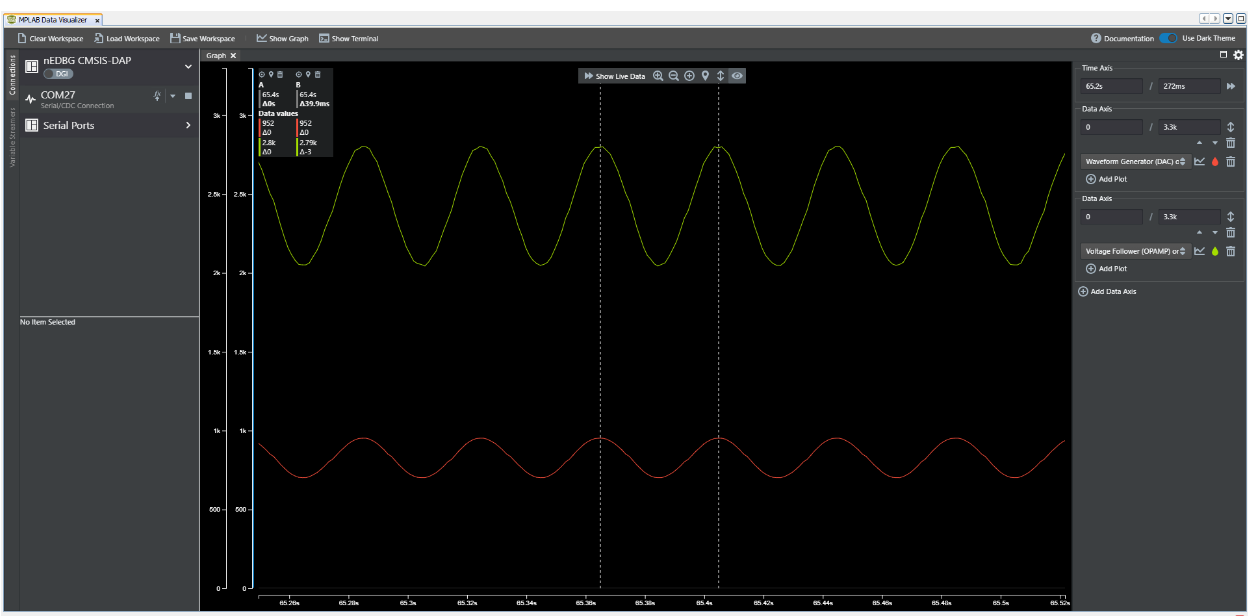
In this section, the AVR DB on the Curiosity Nano will demonstrate the instrumentation amplifier operation, using all three of its internal op amps. Hence the microcontroller is programmed with an application that:

- Configures OPAMP0 as a voltage follower with $V_{DD}/2$ input for the positive terminal
- Uses the on-chip DAC and TCB0 to generate a 25 Hz sine wave, which is fed into the positive input of OPAMP1, configured as a voltage follower
- Sets up OPAMP2 as differential amplifier were the inputs are the output from the OPAMP0 and OPAMP1 voltage followers
- Employs the on-chip TCB1 and ADC for sampling the output of the OPAMP2
- Data streams the ADC result and the DAC sample, via USART, to MPLAB *Data Visualizer*

There are no hardware requirements for this application, as all the above-mentioned modules and connections are internal to the AVR DB.

To get the demo up and running, follow the steps outlined in 6. MPLAB® Data Visualizer. The workspace-file to load is *instrumentation_amplifier.json*, available in the GitHub repository. Figure 11-7 shows an example of the expected result.

Figure 11-7. Instrumentation Amplifier - Data Visualizer Output Graph



The graph shows, in red, the DAC waveform while, in lime green, the sampled output of the OPAMP2, which is the same as the output of the instrumentation amplifier. As expected, from the simulation step, the output signal of the amplifier has been amplified by an order of 3 while keeping the input and output signal in phase.

12. References

More information about the analog signal conditioning block and the tools used in demonstrating its operation modes can be found in the following links:

1. AVR DB product page: www.microchip.com/wwwproducts/en/AVR128DB48.
2. AVR DB Curiosity Nano: www.microchip.com/DevelopmentTools/ProductDetails/PartNO/EV35L43A.
3. MPLAB® Mindi™ Analog Simulator product page: www.microchip.com/mplab/mplab-mindi.
4. Getting Started with the MPLAB® Mindi™ Analog Simulator: www.microchip.com/DS50002564.
5. MPLAB® Data Visualizer Standalone (Windows) : gallery.microchip.com/packages/MPLAB-Data-Visualizer-Standalone(Windows)/.
6. MPLAB® Data Visualizer Software User's Guide : www.microchip.com/DS50003001.

13. Revision History

Doc. Rev.	Date	Comments
A	09/2020	Initial document release

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN: 978-1-5224-6606-2

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-72884388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820