

---

---

## Phase Sequence Detection of Three-Phase AC Supply

---

---

---

### Introduction

---

Author: Srinivasa K R, Microchip Technology Inc.

In three-phase Alternating Current (AC) systems, phase reversal and single phasing, i.e. phase loss, are the commonly observed fault conditions. Phase reversal fault generally arises from human errors during system installation or maintenance, and single phasing fault due to broken wire or mechanical failure, worn contact, blown fuse, and thermal overload. The consequences of these faults are unsafe conditions for systems operation, which can lead to severe damage and failures to the systems as well as injury to personnel. Hence, a phase-sequence and phase-loss monitoring unit is required for the three-phase AC systems. The monitoring unit detects the fault and notifies the system to take subsequent action to protect the load.

This application note describes the implementation of a phase sequence and phase loss detection system for a three-phase AC supply and demonstrates the detection of phase sequence and phase loss faults on emulated three-phase AC systems supply using the AVR128DA48 microcontroller. The demo is based on the AVR128DA48 Curiosity Nano development board, a Curiosity Nano Adapter board, and Mikroelektronika Click boards™. The supplemented firmware is developed with the AVR128DA48 microcontroller, using the Atmel START code configurator and Atmel Studio Integrated Development Environment (IDE). The firmware is also developed with the MPLAB® Code Configurator (MCC) and MPLAB® X Integrated Development Environment (IDE).

The AVR® DA is a feature-rich, high-performance, and low-power 8-bit microcontroller family. The built-in Zero-Cross Detector (ZCD), 12-bit differential Analog-to-Digital Converter (ADC), and timer/counter peripherals are essential in implementing a phase sequence detection application. The AVR DA microcontroller ZCD peripheral features interrupt generation on rising edge, falling edge, and both the edges. The combination of ZCD and the 12-bit differential ADC peripheral provides reliable, efficient, and cost-effective phase sequence detection solution over the conventional non-microcontroller based phase sequence detection solutions available for the multiphase AC supply.

---

### Features

---

This application note features the following contents:

- Overview of the AVR DA microcontroller features used for the three-phase sequence detection applications
- Overview of the three-phase sequence detection application
- A three-phase AC supply emulator
- Demo firmware for phase-sequence detection and phase-loss detection

---

## Firmware

---

The firmware for AVR128DA48 is available on GitHub at the links below:



**View Code Example on GitHub (Atmel Studio)**

Click to browse repository



**View Code Example on GitHub (MPLAB® X)**

Click to browse repository

## Table of Contents

Introduction .....	1
Features.....	1
Firmware.....	2
1. AVR® DA Family Overview.....	5
1.1. Relevant Devices.....	5
1.2. Zero-Cross Detector (ZCD).....	5
1.3. Analog-to-Digital Converter (ADC) .....	6
2. Basics of Phase Sequence Detection .....	7
2.1. Three-Phase Power Supply.....	7
2.2. Phase Sequence Detection and Phase Reversal .....	7
2.3. Phase Loss or Single Phasing Detection .....	8
2.4. RMS Voltage Measurement.....	9
3. Application Overview.....	11
4. Three-Phase Signal Emulation and Data Visualization .....	12
4.1. Three-Phase Sinusoidal Signal Emulation.....	12
4.2. Data Visualizer.....	14
5. Hardware Overview .....	15
5.1. Hardware Tools.....	15
5.2. Hardware Setup.....	16
5.3. Microcontroller Pin Configuration Details.....	17
6. Application Implementation .....	18
6.1. Application Firmware.....	18
6.2. Firmware Modules.....	19
6.3. Microcontroller Peripheral Configuration.....	24
6.4. Program and Data Memory Requirements.....	25
6.5. Software Tools Used .....	26
7. Demo Setup and Operation .....	27
7.1. Demo Setup.....	27
7.2. Data Visualizer Configuration.....	28
7.3. User Inputs.....	33
7.4. Application output indicators.....	34
7.5. Demo Operation.....	35
8. Phase Sequence Detection for Real-Time Applications.....	38
9. Conclusion.....	39
10. References.....	40
11. Appendix .....	41
11.1. Application Implementation with AVR128DA64.....	41

---

---

11.2. Code Configurator.....	42
12. Revision History.....	54
The Microchip Website.....	55
Product Change Notification Service.....	55
Customer Support.....	55
Microchip Devices Code Protection Feature.....	55
Legal Notice.....	56
Trademarks.....	56
Quality Management System.....	57
Worldwide Sales and Service.....	58

## 1. AVR® DA Family Overview

### 1.1 Relevant Devices

The AVR® DA family of microcontrollers are using AVR CPU with hardware multiplier, running at up to 24 MHz, with 32/64/128 KB of Flash sizes, 4/8/16 KB of SRAM, and 512B of EEPROM in 28-, 32-, 48- or 64-pin packages. The family uses the latest technologies from Microchip with a flexible and low-power architecture including Event System and Sleepwalking, accurate analog features, and advanced digital peripherals.

**Figure 1-1. Overview of AVR® DA Family Microcontrollers**

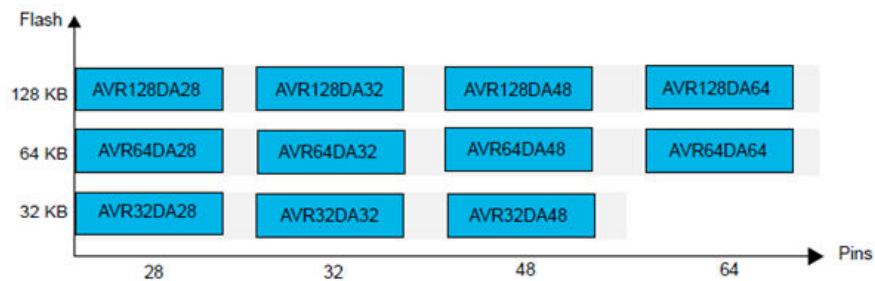


Figure 1-1 shows the AVR DA family microcontrollers with different pin counts and flash memory sizes. The parts with different flash memory sizes typically also have different RAM and EEPROM.

- Vertical migration is possible without code modification, as these devices are fully pin and feature compatible.
- Horizontal migration to the left reduces the pin count and therefore, the available features.

*From the available microcontrollers of AVR DA family, the 48-pin and 64-pin devices are relevant for this application note, as these devices are equipped with two and three instances of ZCD peripheral respectively.*

**Note:** The 48-pin and 64-pin devices of AVR DB family are also relevant for this application note, as these devices are equipped with two and three instances of ZCD peripheral respectively.

### 1.2 Zero-Cross Detector (ZCD)

The Zero-Cross Detector (ZCD) peripheral detects when an alternate voltage crosses through a threshold voltage near to zero volts, i.e. GND potential. The threshold is the zero-crossing reference voltage and is 0.8V above ground for the ZCD peripheral of AVR DA family microcontrollers.

The ZCD peripheral offers more features than the traditional methods used to perform zero-cross detection.

The ZCD peripheral can give the output in the form of a software interrupt on rising edge detection, falling edge detection, and both rising edge and falling edge detection of alternating signal. These features aid users with effective utilization of the ZCD peripheral for detecting zero-crossings on alternating signals.

Additionally, the ZCD peripheral offers the following operational features:

- ZCD logic output indication
- ZCD reference voltage (ZCPINV) offset correction
  - Correction by offset current
  - Correction by AC coupling
  - Handling VPEAK variations
- Event generator to the event system

### **1.3 Analog-to-Digital Converter (ADC)**

The simplest and most commonly-used peripheral for interfacing analog sensors to microcontrollers is the Analog-to-Digital Converter (ADC). The AVR DA family of microcontrollers offers a 12-bit successive approximation register (SAR) ADC, with up to 22 channels, and with a sampling rate of up to 130 ksps at a 12-bit resolution. The input to ADC is fed through a sample-and-hold circuit, which ensures the input voltage to the ADC is held at a constant level during sampling.

The AVR DA family microcontroller supports both Single-Ended and Differential mode ADC operation features.

The ADC is connected to an analog input multiplexer for selection between multiple single-ended or differential inputs. In single-ended conversions, the ADC measures the voltage between the selected input and 0V (GND). In differential conversions, the ADC measures the voltage between two selected input channels. The selected ADC input channels can either be internal (e.g., a voltage reference) or external analog input pins.

## 2. Basics of Phase Sequence Detection

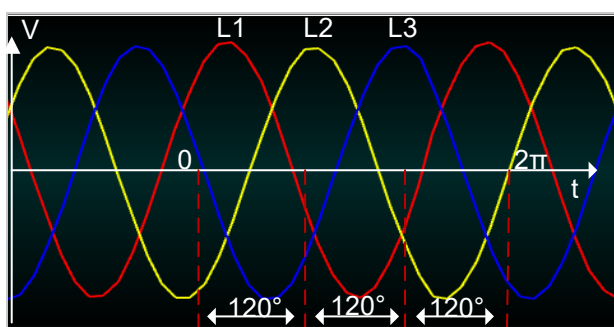
This section gives a brief overview of the three-phase power supply system and its related faults.

### 2.1 Three-Phase Power Supply

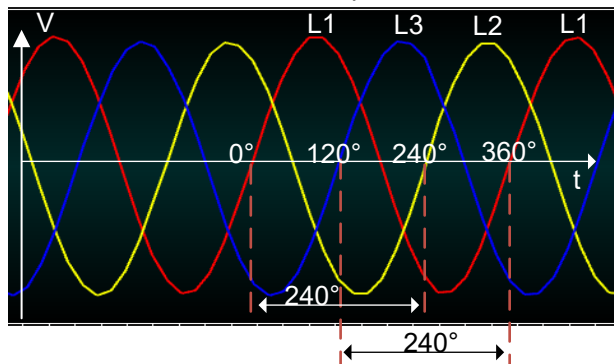
A three-phase power supply system is the most common method of electric power generation, transmission, distribution, and consumption worldwide.

The three-phase power comprises of three alternating phases, generally represented as L1, L2, and L3. All three phase produces AC voltages of equal amplitude and frequency with reference to ground potential. All three phase voltages are phase-shifted to each other by  $120^\circ$ , as shown in Figure 2-1 'a'.

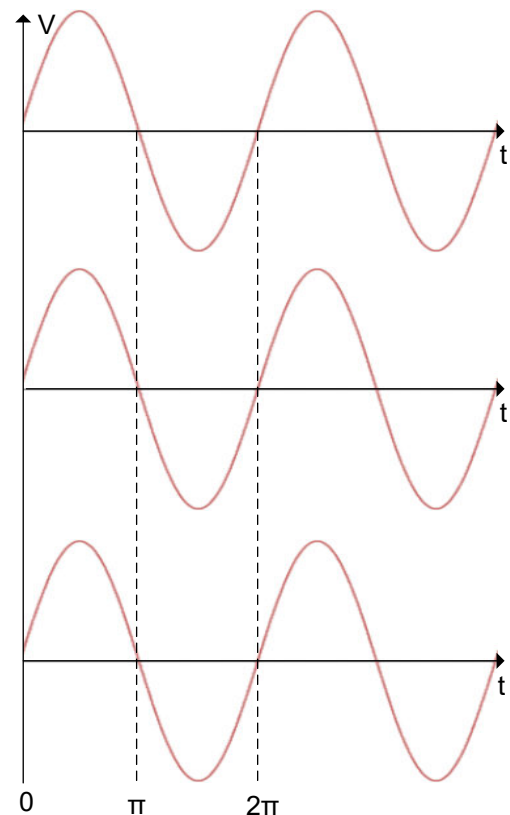
Figure 2-1. Typical Three-Phase Signal



a. Typical Three-Phase Signal with Positive Phase Sequence



b. Three-Phase Power Supply with Negative Phase Sequence



c. Zero Phase Sequence

### 2.2 Phase Sequence Detection and Phase Reversal

The three phases of a three-phase power supply, usually designated as L1, L2 and L3, attain their respective peak voltages periodically one after the other in a particular sequence, and this is referred to as a phase sequence.

The phase reversal occurs when any two phases of a three-phase supply are interchanged from the normal sequence. The phase-reversal fault occurs as a result of mistakes made during equipment installation and commissioning or maintenance.

The phase-reversal fault can be detected by monitoring the phase sequence of the three-phase power supply.

The correct phase sequence is required for driving the load properly in the three-phase AC system. It ensures that the load works as desired. When the phase sequence of the three-phases AC supply is incorrect, the connected

three-phase loads may operate in the opposite direction or result in an unintended operation of the load, thus damaging the system installation.

Three possibilities exist while forming the phase sequence of the three phase supply:

- Positive phase sequence
- Negative or reversed phase sequence
- Zero phase sequence

The phase sequence of the three-phase supply can be monitored by measuring the phase-shift among the three sinusoidal signals, in units of time.

### 2.2.1 Positive Phase Sequence

In case of a positive phase sequence, voltage on all the three lines of three-phase supply cycles through 360° where phase L2 lags phase L1 by 120° and phase L3 lags phase L1 by +240°. The sequence L1 – L2 – L3 is termed as a positive phase sequence.

Figure 2-1 'a' shows the positive phase sequence of three-phase supply, where the phases L1, L2, and L3, attain their respective peak voltages periodically, one after the other in the sequence.

### 2.2.2 Negative Phase Sequence

In case of a negative/reversed phase sequence, the voltage on all the three lines of three-phase supply cycles through 360° while phase L3 lags phase L1 by 120° and phase L2 lags phase L1 by +240°. The sequence L1 – L3 – L2 is termed as negative or reversed phase sequence.

Figure 2-1 'b' shows the negative or reversed phase sequence of the three-phase supply, where phases L2 and L3 are interchanged from the normal sequence.

### 2.2.3 Zero Phase Sequence

If the phases L1, L2, and L3 are parallel to each other, the sequence is referred to as a zero phase sequence.

Figure 2-1 'c' shows the zero phase sequence, where phases L1, L2, and L3 are parallel to each other.

## 2.3 Phase Loss or Single Phasing Detection

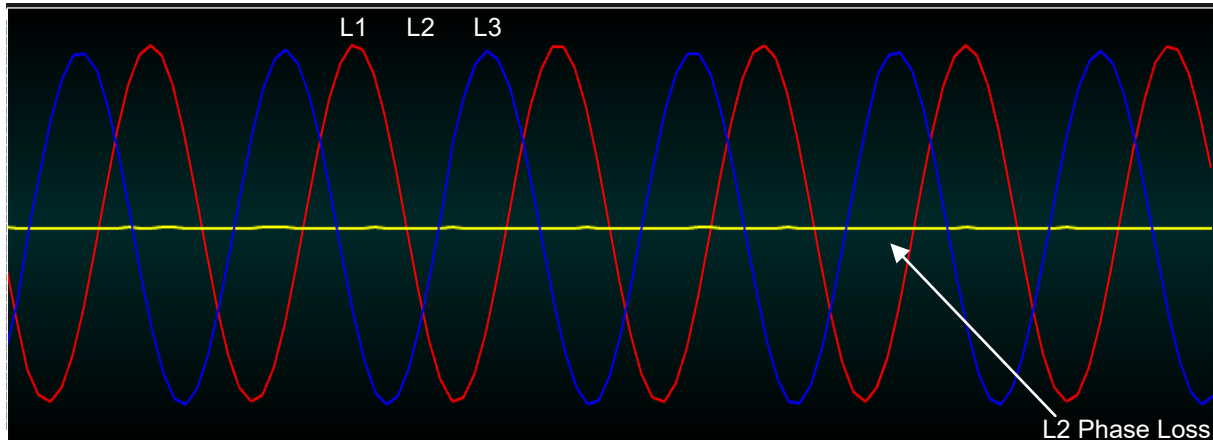
Single phasing, i.e. phase loss, is a very common electrical fault experienced by three-phase systems and it occurs when any phase of the three-phase power supply is lost. The phase loss fault occurs due to blown fuse, thermal overload, broken wire, wire contact or mechanical failure. In case of failure to detect phase loss in the three-phase systems in time, the connected loads and infrastructure can be seriously damaged.

The phase loss of the three-phase supply can be detected either by measuring the Root Mean Square (RMS) voltage of each phase or by monitoring the zero-crossings of the phases using the ZCD peripheral.

If the RMS voltage of any phase of the three- phase supply is 0, it is deemed as phase loss. In case of ZCD based implementation, if the zero-crossing is not detected for any of the phases in a specific time duration, it means loss of the phase.

Figure 2-2 shows the signal phasing on three-phase supply. Phase L2 is lost, phase L3 lags L1 by 240°, and the three-phase signal is 120° out of phase.

Figure 2-2. Three-Phase Signal with L2 Phase Loss



## 2.4 RMS Voltage Measurement

For the AC power supply, RMS is equal to the value of the direct current that would produce the same average power dissipation in a resistive load. The RMS value gives the equivalent DC measurement of the AC power.

Mathematically, the root mean square is defined as the square root of the mean square (the arithmetic mean of the squares of a set of numbers). In alternating signals, RMS can be defined as an integral of the squares of the instantaneous values during a cycle.

In the application, the RMS voltage is calculated by the arithmetic mean of the square of a set of numbers on each phase of the three-phase input signal.

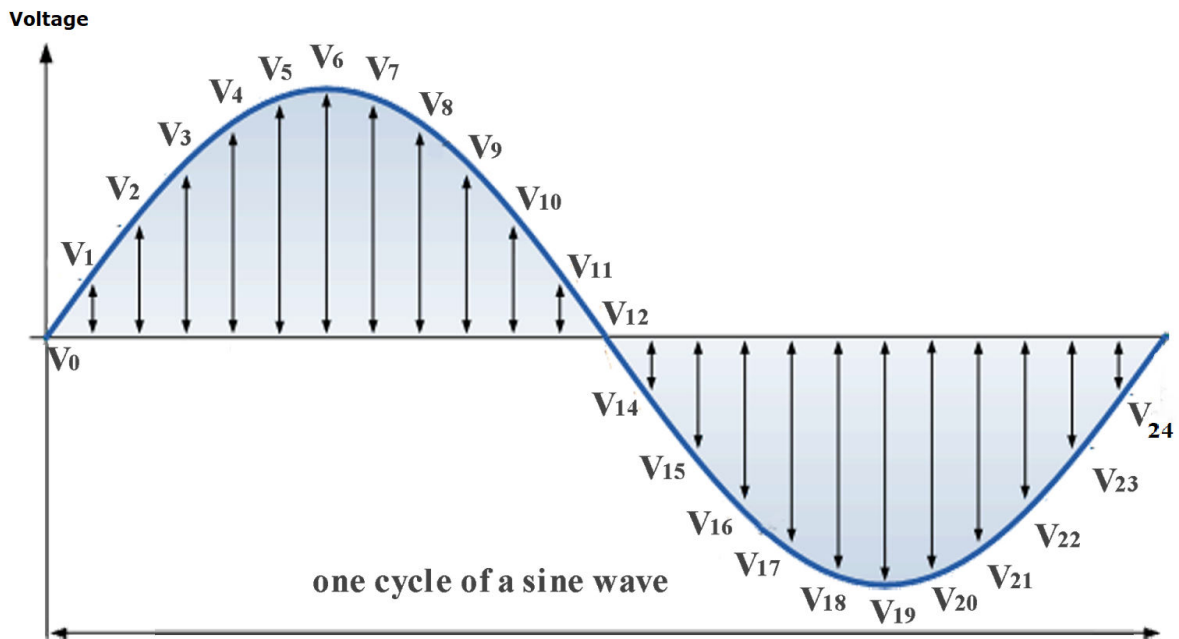
The equations used for calculating RMS voltage are:

- $$V_{rms} = \sqrt{\frac{\text{sum of the squared voltage samples}}{\text{number of samples}}}$$
- $$V_{rms} = \sqrt{\frac{V1^2 + V2^2 + V3^2 + \dots + Vn^2}{n}}$$

Here, n is number of samples and V1,V2,V3.....Vn are the voltage samples acquired from the input signal using ADC.

Figure 2-3 shows the signal, which is divided into 24 number of equal samples.

Figure 2-3. RMS Voltage Calculation Signal



$$V_{rms} = \sqrt{\frac{V_0^2 + V_1^2 + V_2^2 + V_3^2 + V_4^2 + V_5^2 + \dots + V_n^2}{n}}$$

### 3. Application Overview

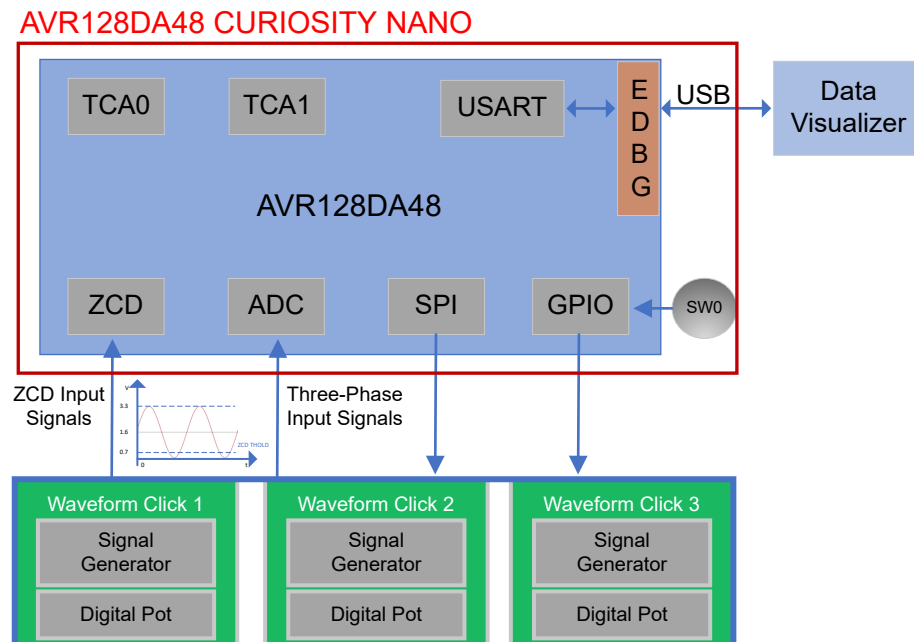
The application demonstrator showcases the phase sequence detection, phase loss detection, and RMS voltage measurement on an emulated three-phase AC supply, using the AVR128DA48 microcontroller.

The demo uses an emulated three-phase AC supply to showcase the essential features of the application. To emulate the AC power supply, three Mikroelektronika waveform clicks are used in the application. The waveform click generates sinusoidal signal with desired amplitude, phase and frequency. The application configures the waveform clicks to generate three 3.3V sinusoidal signal with 50/60 Hz and 120° phase difference to substitute the AC supply. Additionally, to showcase the various faults the waveform clicks can be reconfigured by the application in the run-time to change the output signal.

In the application, the data visualizer tool is used to interact with the application firmware through a UART interface. The data visualizer is used to monitor and report the live status of phase reversal, phase loss, and RMS voltage of the signal. It is also used to send the emulator commands to replicate the various faults by waveform clicks.

Figure 3.1 showcases the block diagram of the application demo used.

**Figure 3-1. Application Overview**



To select between the 50 Hz and the 60 Hz operating mode, the on-board curiosity nano switch is used. The system stays in Frequency Selection mode for 10 seconds after power-up. By default, the system starts with 50 Hz frequency and it can be changed by pressing the switch after the board is powered up. The selection change can be viewed on the dashboard window of the data visualizer.

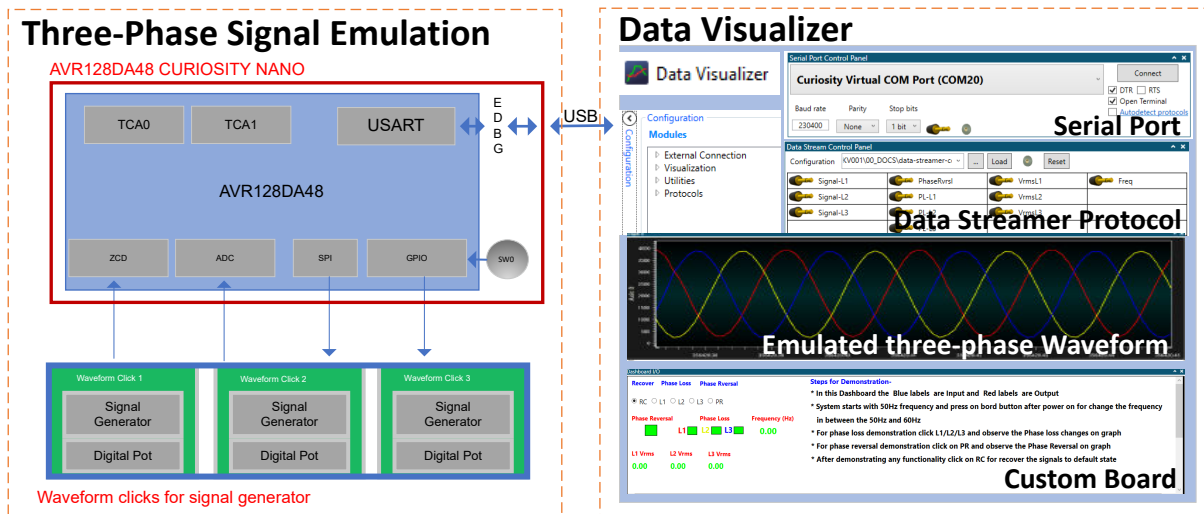
The application continuously monitors the three-phase power supply for phase reversal and phase loss faults. The result of phase sequence, phase loss, and RMS voltage level of the three-phase signal are continuously updated on the dashboard window.

Upon receiving the user inputs to demonstrate phase loss and phase reversal faults from the dashboard, the application updates the configurations of waveform clicks. Thereby, the three-phase signal with desired fault condition is generated by the waveform clicks.

Also, the application restores the fault conditions, when an appropriate command is received from the dashboard to recover from the fault condition.

## 4. Three-Phase Signal Emulation and Data Visualization

Figure 4-1. Block Diagram of the Three-Phase Signal Emulation and Data Visualization



### 4.1 Three-Phase Sinusoidal Signal Emulation

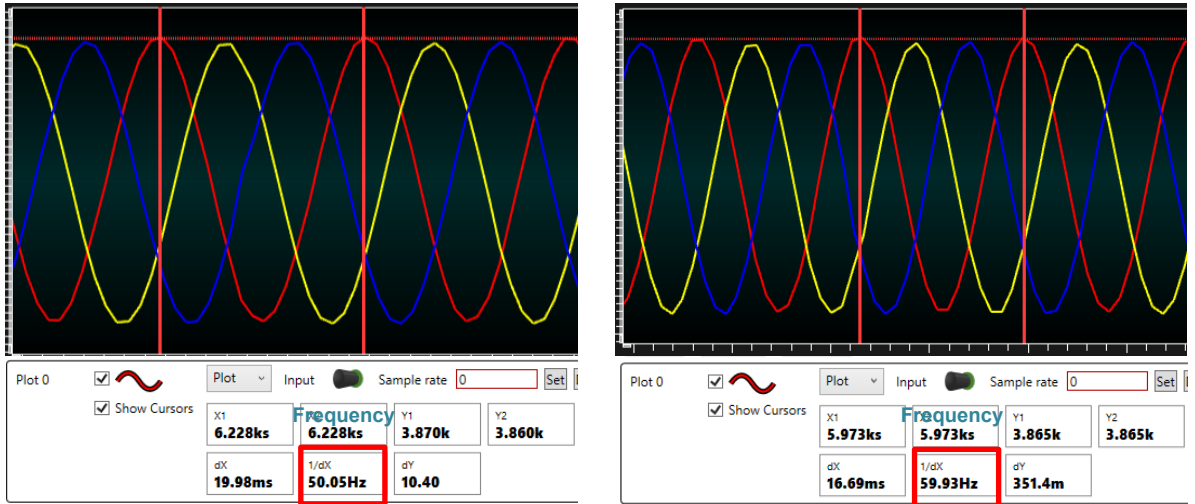
In this application, input to the Phase Sequence Detector is a three-phase AC input. Detection of the phase sequence, phase loss, and RMS voltage calculation of the input supply are the fundamental functionalities of the application.

As discussed in the [Application Overview](#) section, the demo is using a waveform emulator to substitute the three-phase power supply. The following characteristics are required from the emulated signal to demonstrate the application.

- Three-phase signal (L1-L2-L3)
- Configurable amplitude
- Configurable phase-shift
- Configurable frequency

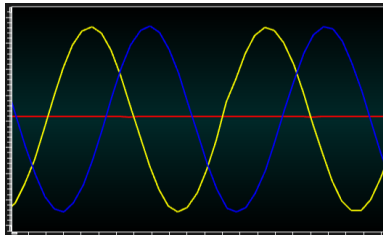
To accomplish the above features a three-phase emulator is realized using waveform click from MikroElektronika. The waveform click can generate the sinusoidal waveform with configurable frequency, phase, and amplitude. Three waveform clicks are used to generate the required three-phase sinusoidal signal with 120° phase-shift which is fed as input to the phase sequence detection system. The emulated three-phase sinusoidal signal with the provision of changing the waveform configurations in run-time helps in demonstrating all the features of implemented Phase Sequence Detection system efficiently.

Figure 4-2. Emulated Signals

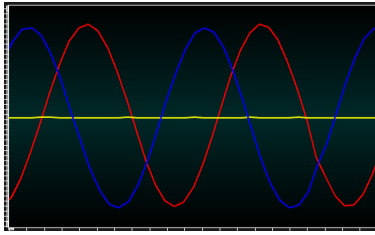


a. Three-Phase emulated signals with 50 Hz

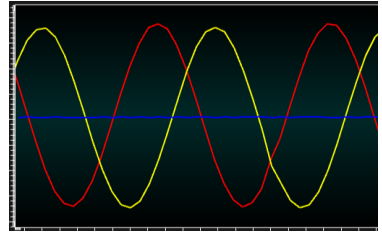
b. Three-Phase emulated signals with 60 Hz



c. Three-Phase emulated signals with L1 phase loss



d. Three-Phase emulated signals with L2 phase loss



e. Three-Phase emulated signals with L3 phase loss

Figure 4-2, 'a' showcases the emulated three-phase signals, which is used for the application demonstration. It consists of three signals with a 50 Hz frequency and a 120° phase shift in between the signals.

Figure 'b' showcases the emulated three-phase signals with a 60 Hz frequency.

Figure 'c' showcases the emulated three-phase signals with L1 phase loss. It consists of three signals with different frequencies: L1 with 0 Hz, L2 and L3 with 50 Hz.

Figure 'd' showcases the emulated three-phase signals with L2 phase loss. It consists of three signals with different frequencies: L2 with 0 Hz, L1 and L3 with 50 Hz.

Figure 'e' showcases the emulated three-phase signals with L3 phase loss. It consists of three signals with different frequencies: L3 with 0 Hz, L1 and L2 with 50 Hz.

## 4.2 Data Visualizer

The data visualizer is a customizable tool (PC software) from Microchip meant to process and visualize data from the microcontroller over a specified communication interface. The data visualizer can receive data from various sources such as the Embedded Debugger Data Gateway Interface (DGI) and/or serial communication ports.

This section gives an overview of the main modules and features of the data visualizer used in this demonstration.

- **Serial Port**

The data visualizer serial port can connect to a PC serial port or USB (using serial to USB). The data visualizer supports all standard baud rates of serial UART communication, and in the application demonstration, the baud rate is set to 230400.

- **Graph**

The graph module is a graph plotting tool. This module allows to plot a number of signals from different sources, such as sliders input, power measurement input, and serial port, simultaneously. This demonstration graph is used to show the three-phase emulator signals.

- **Custom Dashboard**

The dashboard module is a customizable Graphical User Interface (GUI) panel. It can be used to send and receive data from the target application. It can be used to control and display parameters from the application firmware. Elements (button, label, slider, etc.) are placed in the dashboard area to form the GUI. Each element can have an endpoint associated with it to send or receive values. A slider, for example, has a source that outputs the slider position as a numeric value, which can be sent to the application firmware. Endpoints are shown when the Show Endpoints option is selected.

- **Data Stream Protocol**

The data stream module takes an incoming raw data stream and splits it into multiple data streams. The data stream format is specified by a configuration file provided by the user. The configuration file is a comma-delimited text file that specifies one data variable per line.

For more details about the data visualizer, refer to the [Data Visualizer user guide](#).

## 5. Hardware Overview

To perform the demo, development kits from Microchip are used. This aids in the rapid development and deployment of the phase sequence detection application demo.

### 5.1 Hardware Tools

The phase sequence detection demonstrator uses the following hardware:

- [AVR128DA48 Curiosity Nano board](#)

The Curiosity Nano boards allows the easy evaluation of the AVR128DA48 microcontroller. The board also offers full programming and debugging capabilities to support the developers throughout the development process.

- [Curiosity Nano base for Click board](#)

Each Curiosity Nano board is compatible with the Curiosity Nano Base for Click boards. This base includes a socket that fits all Curiosity Nano boards plus three mikroBUS™ sockets and an Xplained Pro extension board connector that will enable the developers to effortlessly expand their designs.

- [Three MikroElektronika waveform clicks](#)

The waveform click consists of a signal generator (AD9833) and digital pot (AD5227). The waveform click can generate various types of signals with frequency up to 12 MHz including sinusoidal waveform.

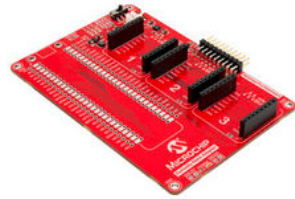
The output signal is available over the SMA connector, allowing the shielded coaxial cable to be used. In this application, a waveform click is used to emulate the three-phase AC supply.

- SMA male cables

Subminiature (SMA) connectors are semi-precision coaxial RF connectors developed as a minimal connector interface for coaxial cable with a screw-type coupling mechanism. SMA cables are used to connect the waveform generated click board output to the curiosity nano.

[Figure 5-1](#) shows the hardware used to realize the application:

**Figure 5-1. Hardware Modules Used**



**Curiosity Nano Base for Click boards™**



**AVR128DA48 Curiosity Nano board**



**Waveform click**

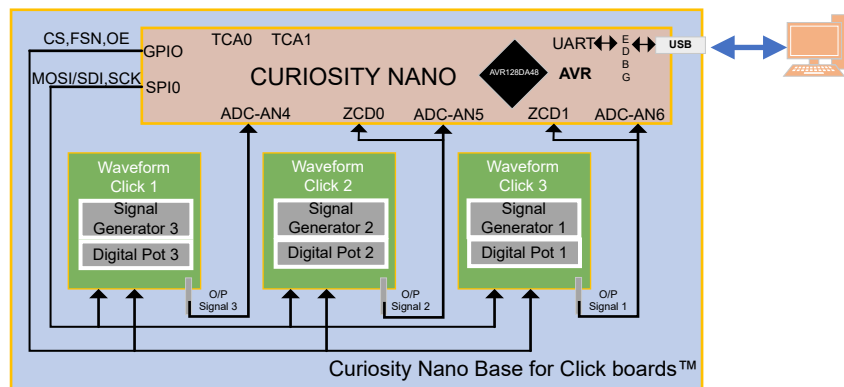


**SMA Male cable**

## 5.2 Hardware Setup

The hardware development boards required for three-phase sequence detection application are connected as shown in [Figure 5-2](#).

**Figure 5-2. Hardware Setup**



The AVR128DA48 Curiosity Nano board is used and it is mounted on a Curiosity Nano base for Click boards.

An SMA male cable is used to connect the waveform clicks output to the respective ZCD input and ADC channel input.

The data visualizer tool is used to demonstrate the phase sequence detection application. Graph view of the data visualizer is used to display the three-phase waveforms generated by the waveform clicks to give a visual feedback of the emulated three-phase power supply system. The custom dashboard feature of the data visualizer is used to

design a dashboard specific for the three-phase sequence detection application. The dashboard shows the current RMS voltage measured by the microcontroller and fault status. It also allows users to give commands to the emulator to emulate any specific fault condition.

The USB-UART bridge interface on the Curiosity Nano board is used to communicate between the data visualizer and the AVR128DA48 microcontroller.

### 5.3 Microcontroller Pin Configuration Details

Below are the pin configuration details of the AVR128DA48 microcontroller for the three-phase sequence detection application.

**Table 5-1. Pin Configuration Details**

Serial No	Microcontroller Pin	Pin Configuration	Peripherals Configuration	Pin Function
1	PA4	SPI-MOSI	SPI	Waveform Click SPI0-MOSI
2	PA6	SPI-SCK	SPI	Waveform Click SPI0-SCK
3	PC0	Rx	USART	USART1 Rx
4	PC1	Tx	USART	USART1 Tx
5	PC7	Input	GPIO	Curiosity Nano Board onboard button
6	PD1	Analog Input	ZCD	ZCD 0 Input
7	PE3	Analog Input	ZCD	ZCD 1 Input
8	PD6	Analog Input	ADC	Waveform Click 1 Output to ADC-AN6
9	PD5	Analog Input	ADC	Waveform Click 2 Output to ADC-AN5
10	PD4	Analog Input	ADC	Waveform Click 3 Output to ADC-AN4
11	PA7	SPI-CS1(Output)	GPIO	Waveform Click 1, AD5227 Chip Selection
12	PD7	SPI-FSN1(Output)	GPIO	Waveform Click 1, AD9833 Chip Selection
13	PD0	OEN1 (Output)	GPIO	Waveform Click 1 Output Enable
14	PE2	SPI-CS2(Output)	GPIO	Waveform Click 2, AD5227 Chip Selection
15	PF3	SPI-FSN2(Output)	GPIO	Waveform Click 2, AD9833 Chip Selection
16	PB5	OEN2 (Output)	GPIO	Waveform Click 2 Output Enable
17	PB4	SPI-CS3(Output)	GPIO	Waveform Click 3, AD5227 Chip Selection
18	PE1	SPI-FSN3(Output)	GPIO	Waveform Click 3, AD9833 Chip Selection
19	PD2	OEN3 (Output)	GPIO	Waveform Click 3 Output Enable

## 6. Application Implementation

The application firmware is developed on a AVR128DA48 microcontroller, generated using:

- Atmel START and Atmel Studio IDE
- MCC and MPLAB X IDE

The implemented firmware is performed with the AVR128DA48 Curiosity Nano development board.

The MCU peripherals, TIMERS, SPI, USART, ADC, and ZCDs are used to implement the functionalities of the phase sequence detection using a three-phase power supply.

The Timer/Counter type A instance 1 (TCA1) is used for scheduling the tasks every 500  $\mu$ s; it is configured to generate an interrupt for every 500  $\mu$ s.

Timer/Counter type A instance 0 (TCA0) is configured to generate an overflow interrupt for every 20 ms. It is used for measuring the time in-between two ZCD interrupts, thereby calculating the phase-shift between the phases of three-phase input.

The SPI is used to integrate the waveform click for generating the three-phase emulated signal; it is configured in Master mode with a 6 MHz SPI clock (SPI\_CLK) and a SPI data mode two.

The USART is used for communication between the host computer and application setup; it is configured with a 230400 baud rate.

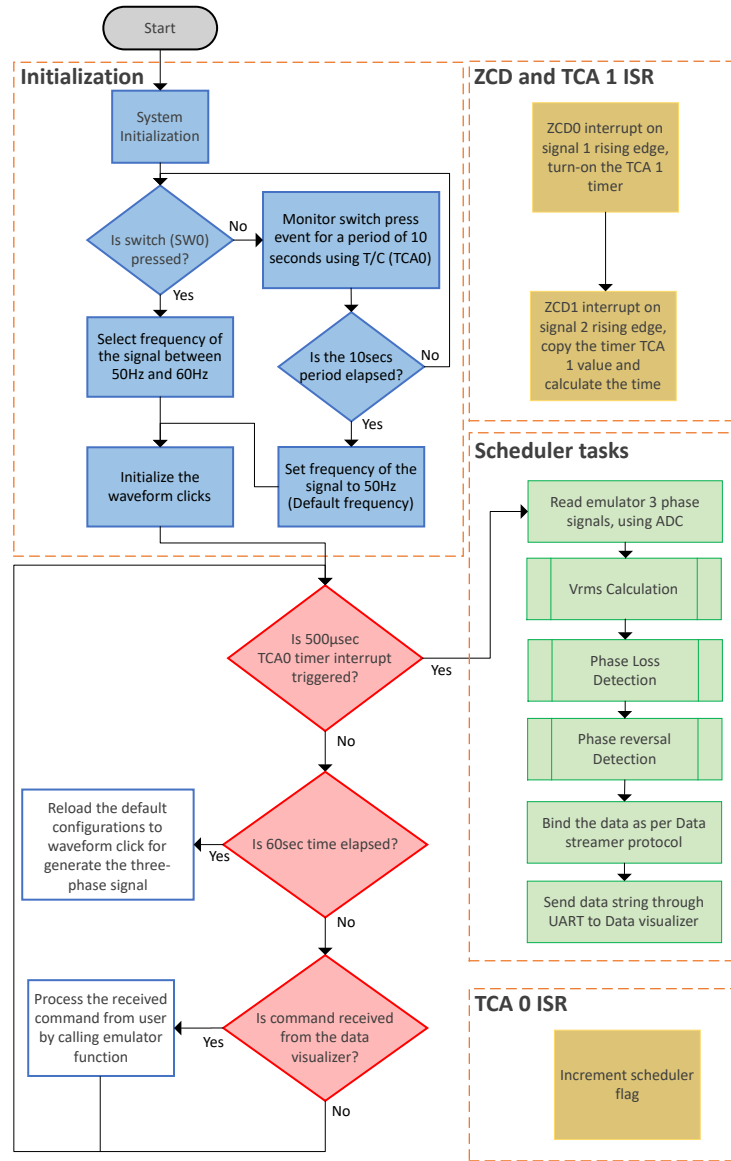
The ADC is used for reproducing the input signal on the data visualizer (for demonstration) and also for three-phase RMS voltage measurement; it is configured with a 2 MHz ADC\_CLK (ADC clock) and 126.3 kHz (0.79  $\mu$ s) sampling rate.

The two ZCD instances are used for detecting the zero cross of the three-phase input signal; they are configured with interrupt on the rising edge signal.

### 6.1 Application Firmware

[Figure 6-1](#) shows the firmware flow of the phase sequence detector application.

Figure 6-1. Application Code Flow



## 6.2 Firmware Modules

Total application firmware is split into five modules:

- Initialization
- Phase reversal detection logic
- Phase loss detection logic
- Task scheduler
- Emulator functionalities

Each module is responsible for a distinctive functionality execution.

### 6.2.1 Initialization

The firmware execution starts with the initialization of the microcontroller and its peripherals. The Phase Sequence Detector application uses microcontroller peripherals such as SPI, TIMER, UART, ADC, Digital I/O, VREF, and ZCD. The system is initialized in the following configurations:

**Table 6-1. Microcontroller Peripherals Configurations**

Peripherals	SPI 0	USART 1	ADC	TCA0	TCA1	VREF	ZCD 0 and ZCD 1
Configurations	SPI Clock (6 MHz)	Baud Rate (230400)	Resolution (12 bits)	TIMER Clock (3 MHz)	TIMER Clock (3 MHz)	Enable reference voltage (ADC)	ZCD enable
	SPI Master mode	Parity bit (0)	ADC Clock (System Clock / 12)	TIMER Mode (Normal)	TIMER Mode (Normal)	Reference Voltage (VDD)	ZCD Interrupt (Rising Edge)
	Data Mode 2	Stop bit (1)	ADC Mode (Differential)	TIMER Period (0xFFFF)	TIMER Period (0x5EC)	—	—

### 6.2.2 Phase Reversal Detection Logic

The ZCD and Timer peripherals are used for calculating the phase-shift between the three-phase signals. The AVR128DA48 device comes with two instances of the ZCD peripheral and both instances are used in this application. The phases L2 and L3 of three-phase signal are connected to the ZCD peripheral to detect zero-crossing. The ZCD peripheral is configured to give an interrupt on the rising edge detection for the signal. Timer peripheral of the microcontroller is used for measuring the period between the L2 phase zero-crossing interrupt and the L3 phase zero-crossing interrupt.

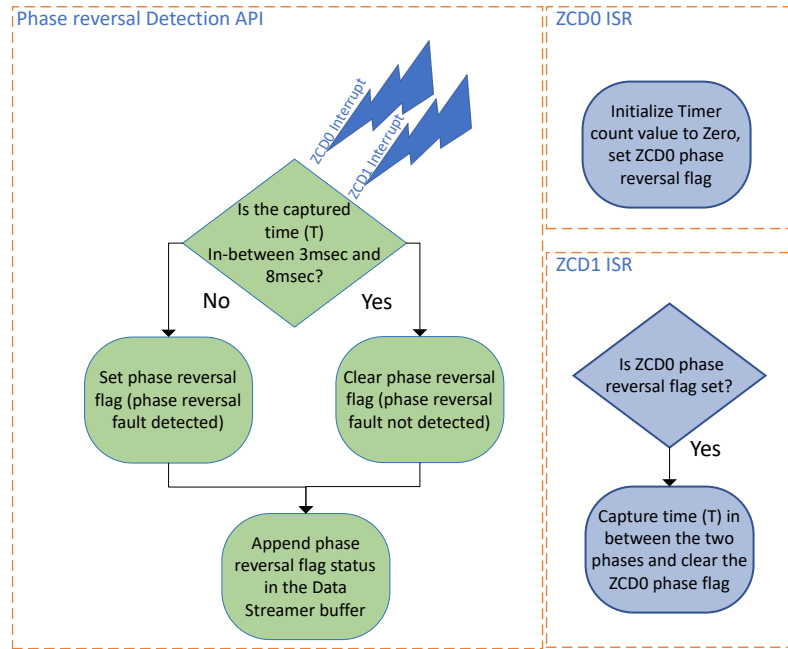
ZCDs are continuously monitoring the phase L2 and L3 of three-phase supply, the phase reversal fault is identified by comparing the time taken from the L2 phase zero-crossing interrupt to the L3 phase zero-crossing interrupt against to time computed on positive sequence three-phase signal.

The calculation provides a method to identify phase shift among the three-phase signals in terms of time for both the positive phase sequence and negative phase sequence conditions.

- The frequency of the three-phase signal is 50 Hz and the phase shift between any two signals of the three-phase signal is 120°
- Period of the signal = 1 / frequency of the signal, i.e. 1/50 = 20 ms
- The time taken to complete a 360° cycle is 20 ms
- The time period between the 120° phase shifted signals of the three-phase signal is 6.66 ms
- **Note:** When the frequency is 60 Hz, the time period between the signals is 5.55 ms
- The time duration between the signals of the positive phase sequence is computed using a general methodology, i.e. L1 to L2 or L2 to L3 or L3 to L1 = 6.66 ms
- When the time duration between the 120° phase-shifted signals of the three-phase signal is 6.66 ms, this is called a positive phase sequence
- The phase reversal fault is detected by comparing the actual time duration between the signals of a three-phase signal with the time duration between the signals of the positive phase sequence

Figure 6-2 shows the phase reversal detection application firmware flow.

Figure 6-2. Phase Reversal Detection



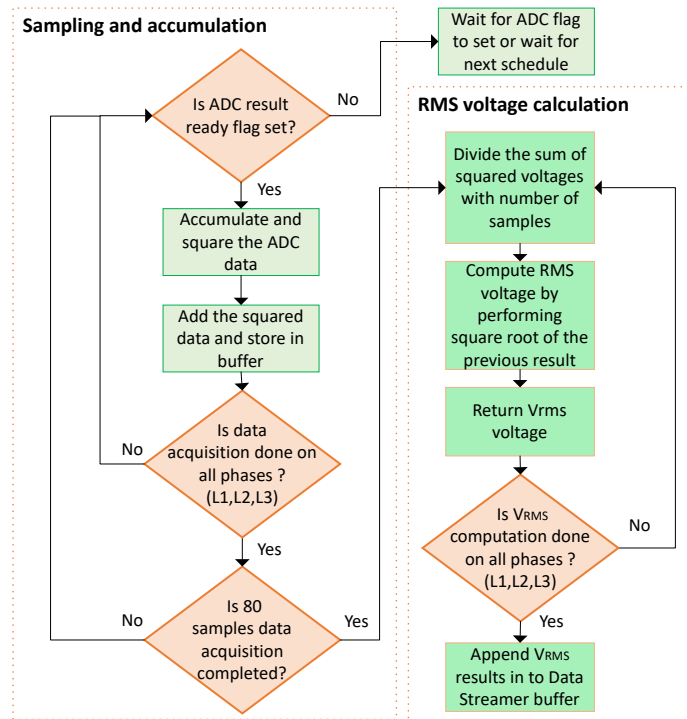
6.2.3 Phase Loss Detection Logic

The RMS voltage measurement accurately measures the voltages of the three-phase sinusoidal signals and ensures accurate monitoring of the three-phase sinusoidal signals. This will help in identifying a single-phase fault condition.

The equation used for RMS voltage calculation is:  $V_{rms} = \sqrt{\frac{V1^2 + V2^2 + V3^2 + ..... + Vn^2}{n}}$

If the computed RMS voltage on any of the phases is zero volt, it means that the corresponding phase is lost.

Figure 6-3. RMS Voltage Calculation Flow Diagram



The ADC peripheral is used for calculating the RMS voltage. The AVR128DA48 device has a 12-bit ADC and it is used in this application with Differential mode configuration.

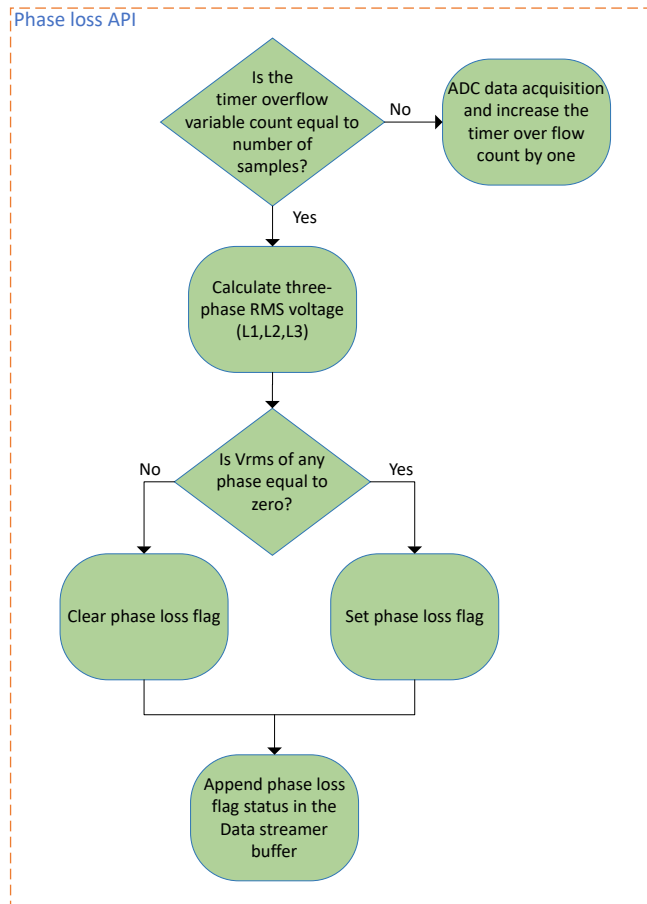
The waveform clicks generate emulated three-phase sinusoidal signal with the amplitude in the range between 0 to 3.3V. The phases L1, L2, and L3 of three-phase signal are connected to the ADC channels, while the ADC measures the instantaneous value of the AC signal and gives a digital equivalent value.

**Note:** In case of the AC mains power supply, the three-phase sinusoidal signal of the power supply must be scaled down and level shifted to conform to the ADC input range requirements.

The application firmware takes 80 instantaneous samples of the 50 Hz AC signal and 64 instantaneous samples of the 60 Hz AC signal. The RMS voltage value is then calculated on these samples by taking the integral of their squares. [Figure 6-3](#) shows the firmware flow diagram of the RMS voltage measurement.

[Figure 6-4](#) gives the firmware flow for the phase loss calculation API. Calculated RMS voltage level is used to check for the phase loss event.

Figure 6-4. Phase Loss Flow Diagram



### 6.2.4 Task Scheduler

The Timer/Counter Type A (TCA1) schedules the task for every 500  $\mu$ s to execute the below functionalities:

- Measure the emulator three-phase signal
- $V_{rms}$  calculation
- Phase loss detection
- Phase reversal detection
- Bind and generate the data array, as per the data stream protocol

In the application, the data visualizer is used for displaying the three-phase AC signal graphical view, measured RMS data, and Fault status. The three phases (L1, L2, and L3) of emulated three-phase signal are fed as input to the ADC peripheral. The converted digital data is transmitted to the data visualizer over a USB-UART interface, to reproduce the emulated three-phase signal on data visualizer.

The data streamer array consists of instantaneous digital values of the three-phase emulated signal, RMS voltage level, frequency measured, and fault status. The application output data is appended into the end of data streamer data array, according to Atmel [data streamer](#) protocol.

The user can send commands from the data visualizer to demonstrate features of the three-phase Sequence detector. The microcontroller receives user commands from the data visualizer through a USB-UART interface for changing amplitude (to demonstrate phase loss) and phase (to demonstrate phase reversal) of the three-phase signal in run time.

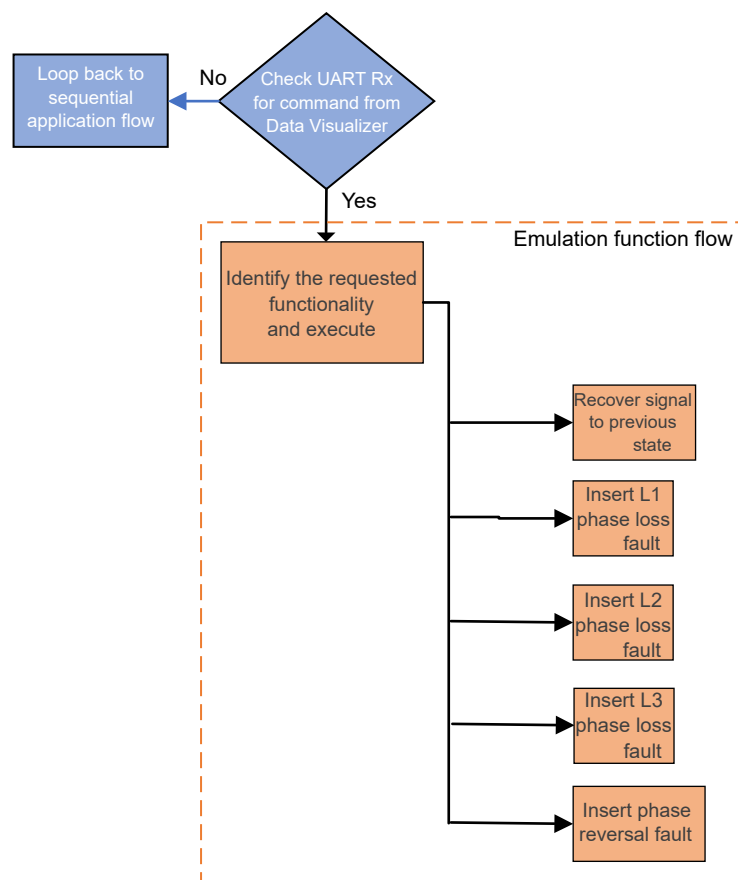
### 6.2.5 Emulator Functionalities

The emulator functionalities section in the application flow diagram processes the user commands required for application demonstration. The below list shows the functionalities processed by the emulator functionalities section in firmware:

- Emulate no-fault three-phase signals
- Emulate phase-loss fault by losing the L1 phase
- Emulate phase-loss fault by losing the L2 phase
- Emulate phase-loss fault by losing the L3 phase
- Emulate phase-reversal fault by changing the phase sequence to L1-L3-L2
- Recover the three-phase signal from the faults and generate the signal with the prior configuration values

Emulator functionalities are executed by sending the desired configuration values to waveform clicks through the SPI serial interface.

**Figure 6-5. Emulator Flow Diagram**



**Note:** The firmware solution is also implemented with AVR128DA64 device. In this solution, the three instances of the ZCD peripheral are used for phase-loss detection. For more details about the implementation, refer to the [Appendix](#) section.

### 6.3 Microcontroller Peripheral Configuration

[Table 6-2](#) details the operating conditions of the AVR128DA48 microcontroller and the various peripherals configured.

**Table 6-2. Microcontroller Peripheral Configuration**

Serial No	Parameter	Value
1	MCU	AVR128DA48
2	MCU Clock	24 MHz
3	MCU Operating Voltage	3.3V
4	Interrupt scheduling configuration	Round robin scheduling
5	Priority Configuration- LVL0PRI	1
6	Interrupt Vector Priority- LVL1VEC	32
7	SPI	SPI 0
8	SPI Clock	6 MHz
9	SPI Mode	2
10	USART	USART 1
11	USART Baud Rate	230400
12	TIMER	TCA0, TCA1
13	TIMER Clock	3 MHz
14	ADC Mode	Differential
15	VREF	VDD to ADC
16	ZCD	ZCD0, ZCD1
17	ZCD Interrupt	Interrupt on Rising edge

The Atmel START Code configurator and MPLAB Code Configurator are used to generate seamless, easy-to-understand C code to be inserted into application firmware, refer to the [Appendix](#) section for more information.

## 6.4 Program and Data Memory Requirements

Here are the application firmware program and data memory requirements with Atmel Studio and MPLAB X IDEs:

**Table 6-3. Memory Requirements with Atmel Studio and Various Compiler Optimizations**

Serial No	Optimization	Program Memory in bytes	Data Memory in bytes
1	-O0	13736 (10.5%)	137 (0.8%)
2	-O1	5136 (3.9%)	137 (0.8%)
3	-Os	5018 (3.8%)	137 (0.8%)

**Table 6-4. Memory Requirements with MPLAB X and Various Compiler Optimizations**

Serial No	Optimization	Program Memory in bytes	Data Memory in bytes
1	0	12878 (10%)	225 (1%)
2	1	5516 (4%)	148 (1%)
3	s	5422 (4%)	148 (1%)

### 6.5 Software Tools Used

Microchip's IDE, compiler, and graphical code generators are used throughout the application firmware development to provide an easy and hassle-free user experience. The following AVR tool chain is used for the demo application:

#### **MPLAB X Platform:**

- MPLAB X IDE (v5.40)
- XC8 Compiler (v2.20)
- MPLAB Code Configurator (MCC) (v3.95)
- AVR 8-bit MCU library (v2.2.0)
- Foundation Services library (v0.2.0)
- Standalone Data Visualizer (v2.20.674)

#### **Atmel Studio Platform:**

- Atmel Studio (v7.0.2397)
- AVR GCC compiler (v5.4.0)
- Atmel Start (v1.7.391)
- Standalone Data Visualizer (v2.20.674)

**Note:** For running the demo, the tool versions installed should be identical with the ones mentioned above, or of later versions. This example is not tested with any of the previous versions.

## 7. Demo Setup and Operation

### 7.1 Demo Setup

The following steps explain the procedure for setting up the hardware for Three-Phase Sequence Detection using the AVR128DA48 microcontroller.

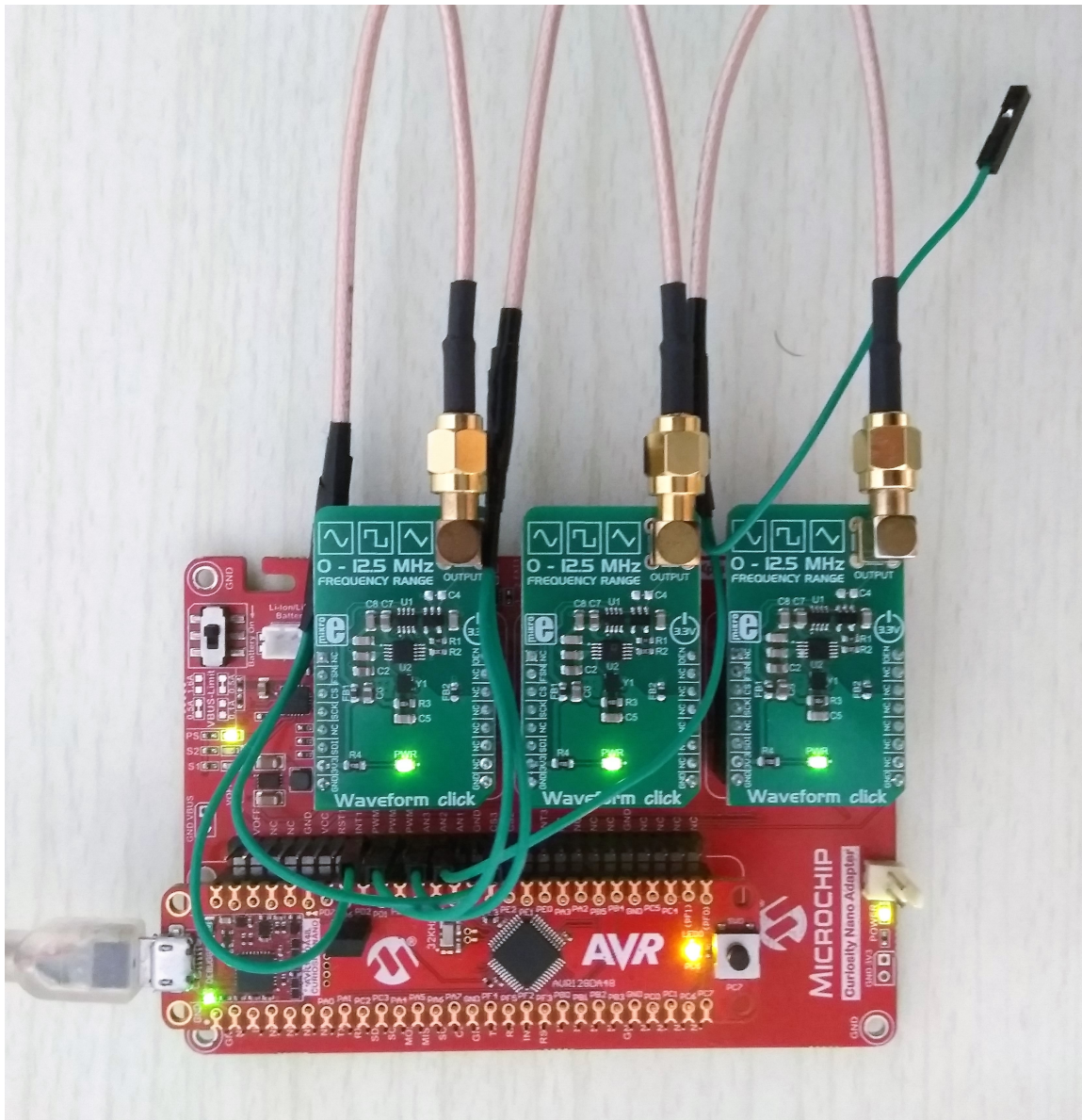
1. Insert the AVR128DA48 Curiosity Nano board in the Curiosity Nano slot of Curiosity Nano base for Click boards.  
**Note:** For proper orientation while inserting the Curiosity Nano board, refer to the marking on Curiosity Nano base board.
2. Insert three waveform clicks in the three mikroBUS slots (slot 1, slot 2, and slot 3) available on Curiosity nano base board.
3. Connect SMA output of the waveform clicks to the respective microcontroller I/O pins using SMA connectors, as shown in the [Table 7-1](#).
4. Disconnect the waveform click 2 OEN pin from the mikroBUS slot2 PWM2 pin.
5. Disconnect the waveform click 3 CS pin from the mikroBUS slot3 CS3 pin.
6. Connect the PWM2 pin of the mikroBUS slot2 to pin PB5, and the CS3 pin of the mikroBUS slot3 to pin PB4.
7. Connect the Curiosity Nano board to a host computer (PC) using the standard micro B USB cable.
8. Refer to the [Firmware](#) section for firmware and hex file.
9. To program application firmware on the microcontroller, refer to the [AVR128DA48 Curiosity NanoHardware User Guide](#).

**Table 7-1. Connection Between the mikroBUS Slots and the Microcontroller**

Serial no	mikroBUS pin	Microcontroller pin
1	Waveform Click 1 output signal connector (SMA)	PD6, PE3
2	Waveform Click 2 output signal connector (SMA)	PD5, PD1
3	Waveform Click 3 output signal connector (SMA)	PD4
4	mikroBUS slot 2 PWM2 pin	PB5
5	mikroBUS slot 3 CS3 pin	PB4

[Figure 7-1](#) shows the hardware setup required for the three-phase sequence detector demonstration.

Figure 7-1. Demonstration Setup using the AVR128DA48 Curiosity NANO Board



## 7.2 Data Visualizer Configuration

The data visualizer tool is used as a graphical user interface while demonstrating the application by displaying the emulated three-phase supply waveforms, output status, and for receiving the user inputs.

Pre-configured data streamer file and custom dashboard file are available with the firmware package. The below steps given for data visualizer configuration use these files to prepare the setup.

**Note:** Both these files are attached to phase-sequence-detection-of-ac-supply GitHub example. For more details refer the [Firmware](#) and [Application Implementation with AVR128DA64](#) sections in the appendix.

The data visualizer modules used for demonstration:

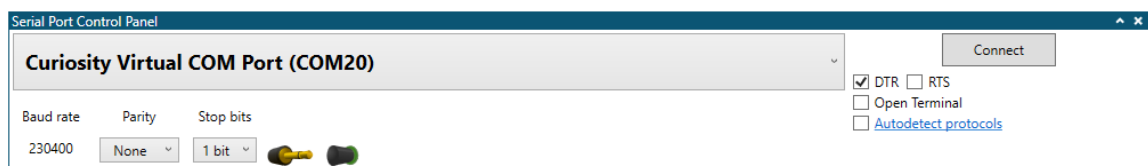
- Serial Port
- Graph
- Custom Dashboard

- Element Label
- Element Signal
- Radio Group
- Data Streamer

### Data Visualizer Serial Port Configuration

- Configure the serial port for communication with the Curiosity Nano board.
- To open Serial Port control panel window, open the **Configuration** tab located on the left side of the data visualizer tool, expand **External Connection** and click the **Serial Port, Serial Port Control Panel window that appears**, as shown in [Figure 7-2](#).
- Set the baud rate to 230400, parity to none and stop bits to 1.
- Ensure the **DTR** option is checked and both **RTS** and **Open Terminal** options are unchecked, as shown in [Figure 7-2](#).

**Figure 7-2. Serial Port Window**



### Data Streamer Configuration

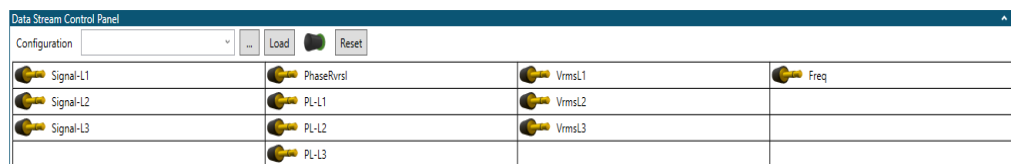
- Configure the data streamer for interfacing, the Graph window, and Dashboard window with the serial port control panel.
- To open data stream control panel, open the **Configuration** tab located on the left side of the data visualizer tool, expand **Protocols** and click the **Data Streamer, Data Streamer** window that appears, as shown in [Figure 7-3](#).
- Click the '...' symbol in the Data Stream control panel window, set an appropriate path for the provided data streamer file from the host computer, as shown in [Figure 7-3](#).

**Figure 7-3. Data stream Control Panel**



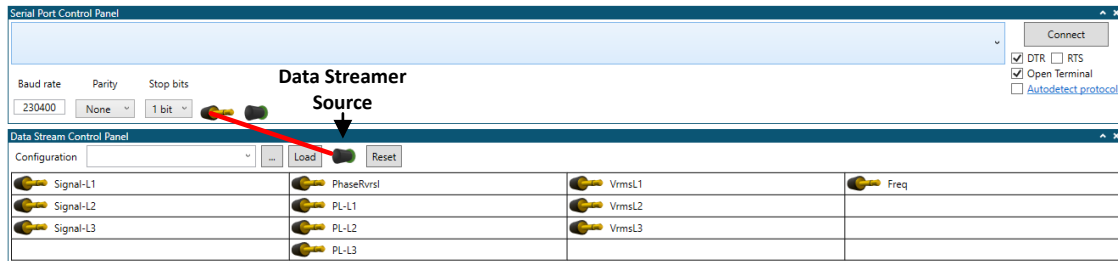
- Click **Load** in the data streamer control panel window, as shown in [Figure 7-3](#). The Data Streamer window will load with ports, as shown in [Figure 7-4](#).

**Figure 7-4. Data Stream Control Panel with Ports**



- Drag **COM0 output** of the Serial Port control panel window and drop on to **Source port** of data streamer control panel window, as shown in [Figure 7-5](#).

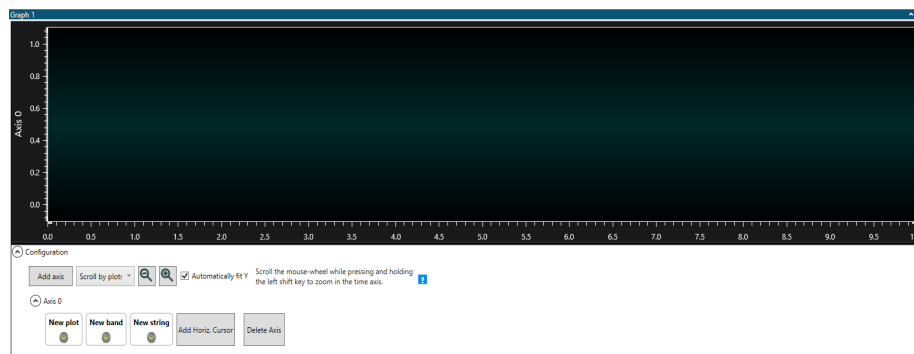
Figure 7-5. Connections between the Serial Port and Data Stream Windows



### Graph Configuration

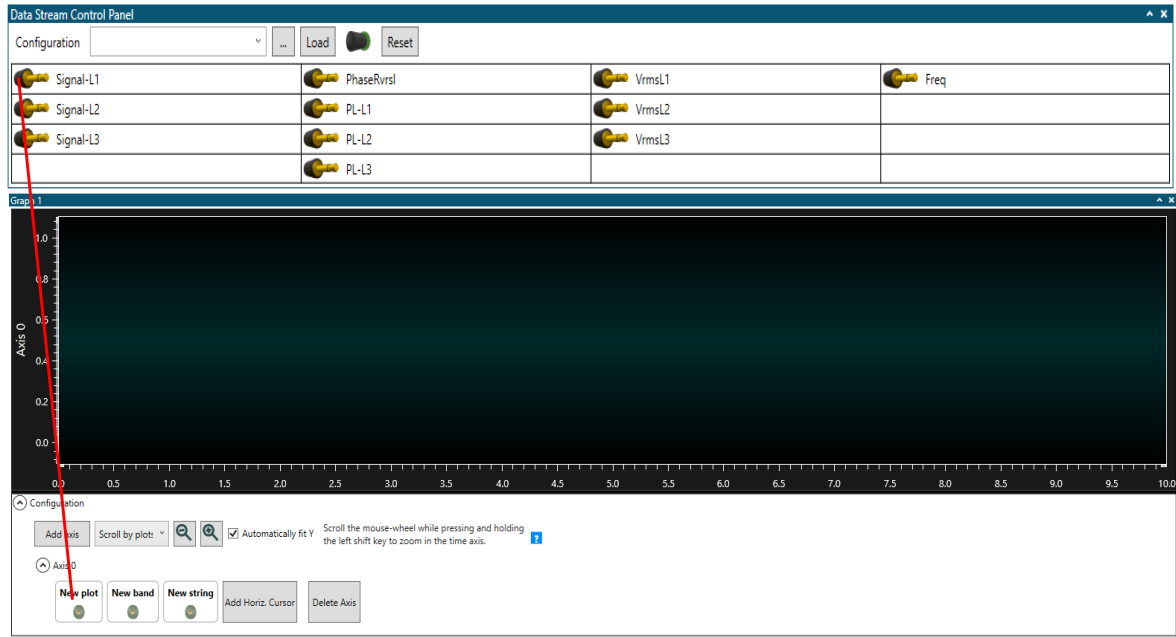
- Configure the Graph window for displaying the emulated three-phase supply AC waveforms, generated by the waveform clicks.
- To open Graph window, click the **Configuration** tab located on the left side of the data visualizer tool, expand **Visualization** option and click **Graph**, **Graph module that appears**, as shown in [Figure 7-6](#).

Figure 7-6. Graph module



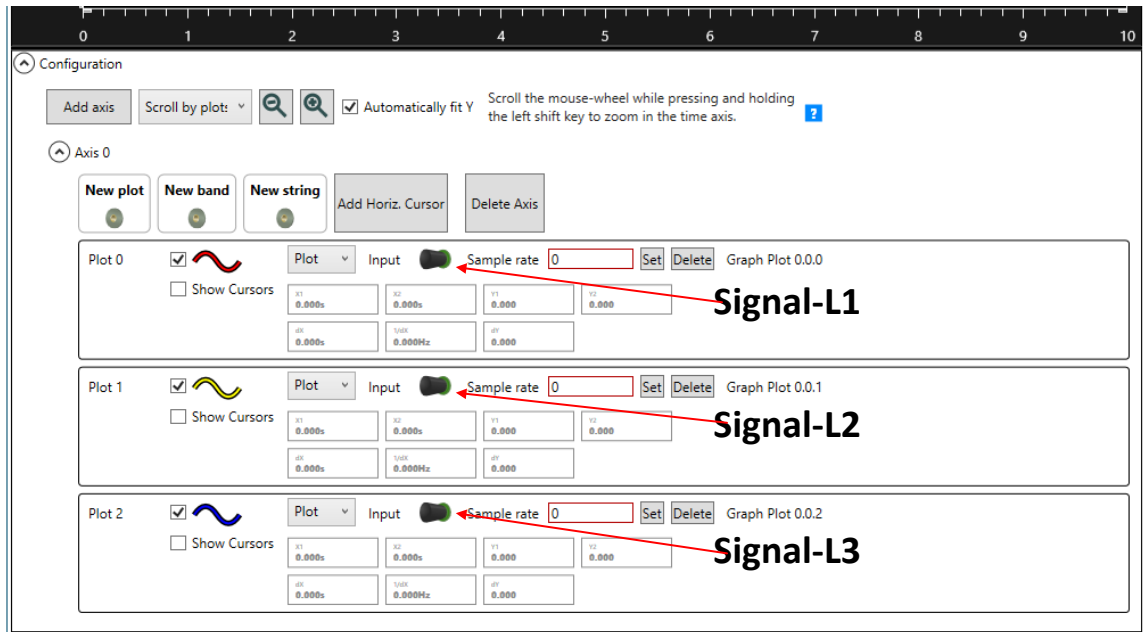
- The required connections between the data stream control panel and the graph window are explained below.
  - Drag Signal-L1 port from Data Stream control panel and drop onto New Plot below the Axis 0 label in Graph window.
  - Drag Signal-L2 port from Data Stream control panel and drop onto New Plot below the Axis 0 label in Graph window.
  - Drag Signal-L3 port from Data Stream control panel and drop onto New Plot below the Axis 0 label in Graph window.
- [Figure 7-7](#) shows the reference figure for connection between Data stream and Graph.

**Figure 7-7. Connections between Graph and Data Stream Windows**



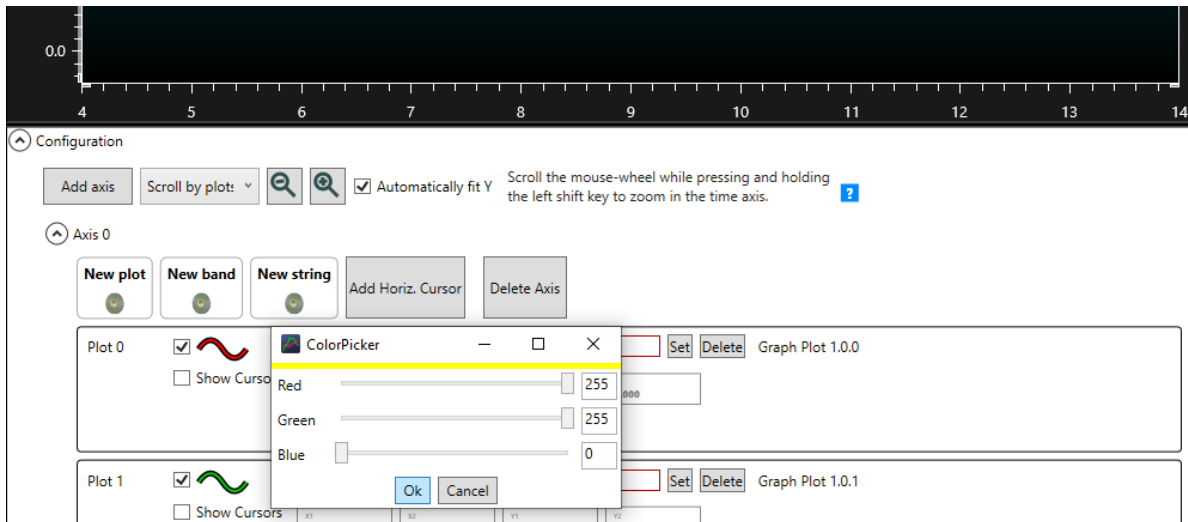
- After the connections are done between the data streamer and the Graph module, the Graph module configuration window appears, as shown in [Figure 7-8](#).

**Figure 7-8. Graph Module Configuration Window**



- Refer to [Figure 7-12](#) to cross verify the connections between the data stream control panel and the Graph window.
- Double click **Plot 1 sine wave symbol** and adjust the (R, G, B) color index values to (255, 255, 0) to set the yellow color, as shown in [Figure 7-9](#).

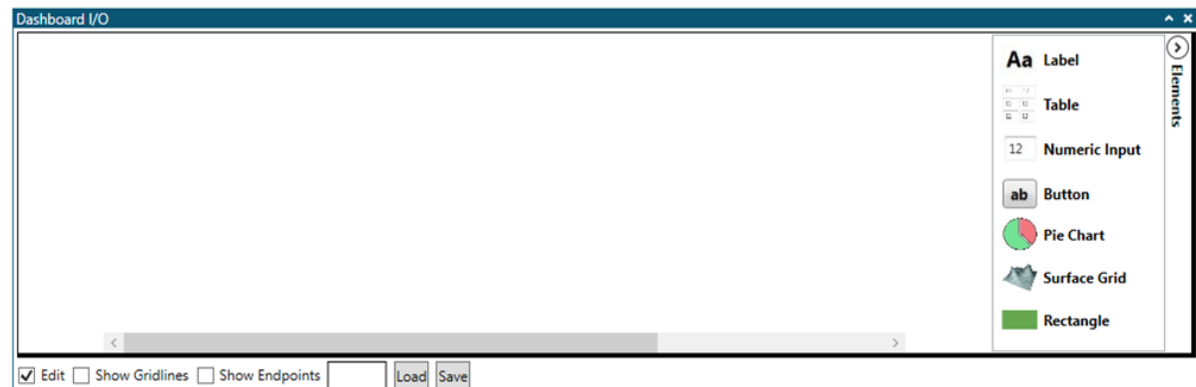
Figure 7-9. Signal Color Selection in Graph Window



### Custom Dashboard Configuration

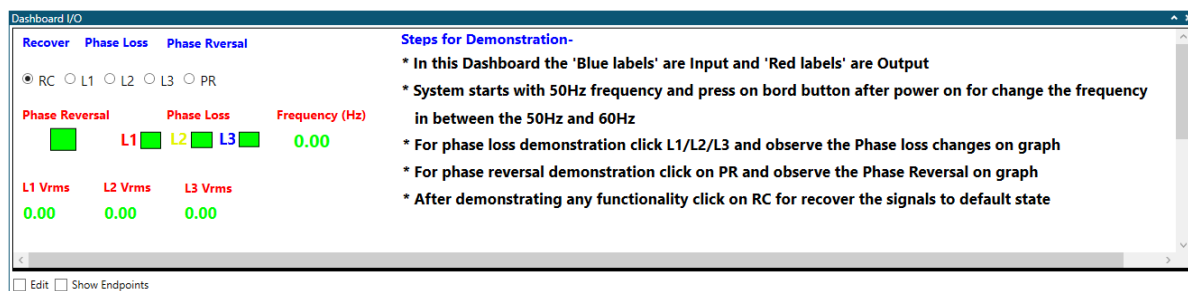
- Configure the custom dashboard window to display the output and to receive user inputs.
- To open the custom dashboard window, open the **Configuration** tab located on the left side of the data visualizer tool, expand **Visualization**, and click **Custom Dashboard** that appears as shown in [Figure 7-10](#).

Figure 7-10. Custom Dashboard Window



- Check **Edit** below the dashboard I/O window, click **Load**, and set the appropriate path for the provided custom dashboard file from the host computer, as shown in [Figure 7-11](#).

Figure 7-11. Custom Dashboard after Loading the Dashboard File

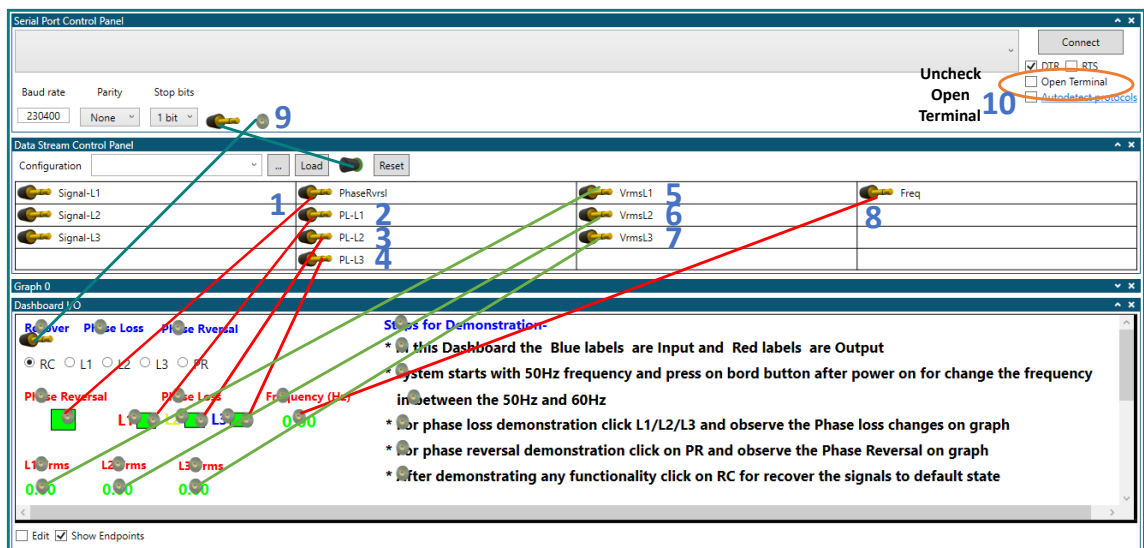


- Check **Show Endpoints** below the dashboard I/O window.

- The required connections between data stream control panel and dashboard I/O window are explained below:
  1. Drag the PhaseRvrsl port from the Data Stream control panel and drop onto the square green box below the text Phase reversal in the dashboard I/O window.
  2. Drag the PL-L1 port from Data Stream control panel and drop onto the green square box below the text Phase Loss and next to the label 'L1' in the dashboard I/O window.
  3. Drag the PL-L2 port from Data Stream control panel and drop onto the green square box below the text Phase Loss and next to the label 'L2' in the dashboard I/O window.
  4. Drag the PL-L3 port from Data Stream control panel and drop onto the green square box below the text Phase Loss and next to the label 'L3' in the dashboard I/O window.
  5. Drag the VrmsL1 port from Data Stream control panel and drop onto the 0.00 value (highlighted in green) below the L1  $V_{rms}$  label in the dashboard I/O window.
  6. Drag the  $V_{rms}L2$  port from Data Stream control panel and drop onto the 0.00 value (highlighted in green) below the L2  $V_{rms}$  label in the dashboard I/O window.
  7. Drag  $V_{rms}L3$  port from Data Stream control panel and drop onto 0.00 value (highlighted in green) below the L3  $V_{rms}$  label in the dashboard I/O window.
  8. Drag the Freq port from Data Stream control panel and drop onto the 0.00 value (highlighted in green color) below the Frequency (Hz) label in the dashboard I/O window.
  9. Connect the Radio button array in the dashboard I/O and below the label Recover (blue label) to the Serial Port control panel receiver port, as shown in Figure 7-12.
  10. Uncheck the **Show Endpoints** radio button below the dashboard I/O, after the connections are completed.

Figure 7-12 shows the required connections between the custom dashboard, data stream control panel, and the serial port control panel windows.

Figure 7-12. Data Visualizer Connection Reference Diagram



## 7.3 User Inputs

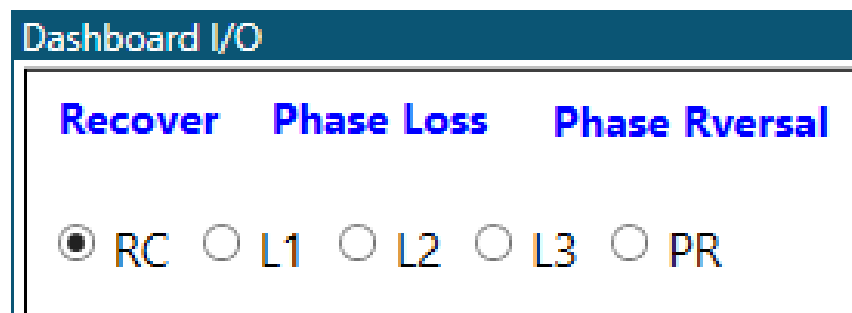
This section explains the list of user input options available on custom dashboard window of data visualizer for demonstrating phase reversal and phase loss faults.

- Recover three-phase supply from faults:
  - Select **RC** to recover the three-phase signal from either the phase loss fault or the phase reversal fault.
- Insert phase-loss fault on L1 phase:

- Select **L1** to emulate L1 phase-loss fault on three-phase signal, which disables the L1 phase out of the three-phase signal.
- Insert phase-loss fault on L2 phase:
  - Select **L2** to emulate L2 phase-loss fault on three-phase signal, which disables the L2 phase out of the three-phase signal.
- Insert phase-loss fault on L3 phase:
  - Select **L3** to emulate L3 phase-loss fault on three-phase signal, which disables the L3 phase out of the three-phase signal.
- Insert phase-reversal fault:
  - Select option **PR** to emulate the phase-reversal fault on three-phase signal, which changes the sequence of the three-phase signal to L1-L3-L2.

Figure 7-13 shows the user input options for phase sequence detection demonstration.

Figure 7-13. User Input Options on Dashboard I/O Window

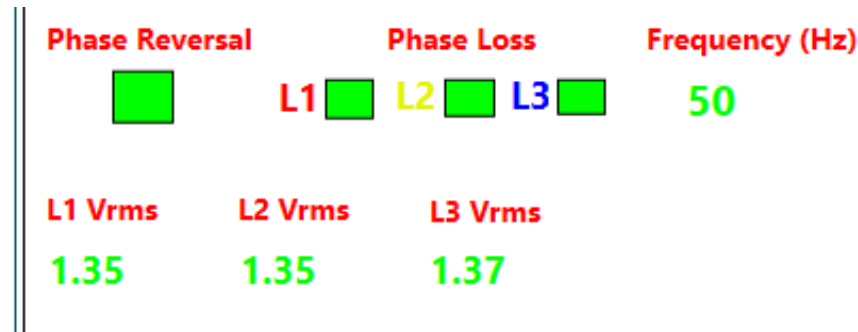


## 7.4 Application output indicators

- Phase-reversal status indicator:
  - The application updates the phase-reversal status in run time. The phase-reversal status indicator shows the color green during the positive phase sequence and the color red, upon detection of phase-reversal fault.
- Phase-loss status indicator:
  - The application updates phase-loss status in run time. The phase loss status indicator shows the color green when there is no phase loss fault and shows the color red, upon detection of phase-loss fault.
- L1 Vrms, L2 Vrms, and L3 Vrms values:
  - The application measures the RMS voltage of all the three phases in run time. The measured RMS voltage values are displayed on dashboard window.
- Operating frequency (Hz):
  - Display the frequency of the three-phase signal.

Figure 7-14 shows the application output status indicator for the phase sequence detector demonstration.

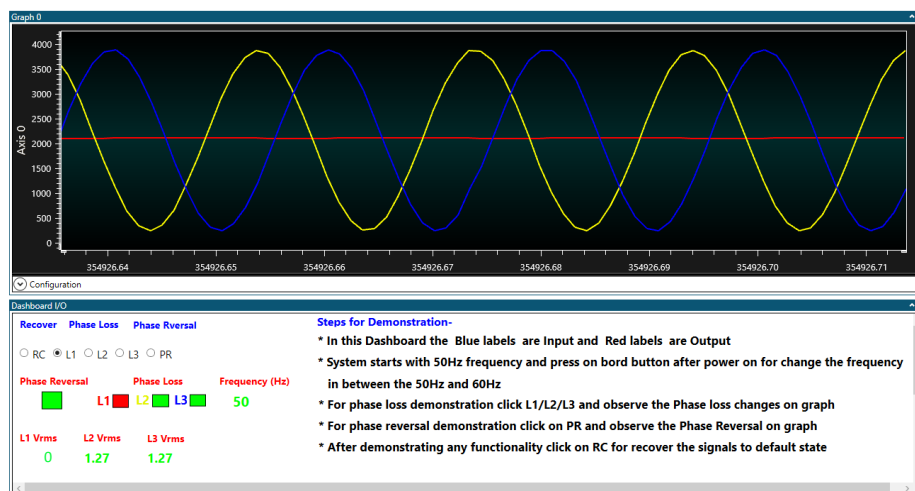
Figure 7-14. Application Output Status Indicator on Dashboard I/O Window



## 7.5 Demo Operation

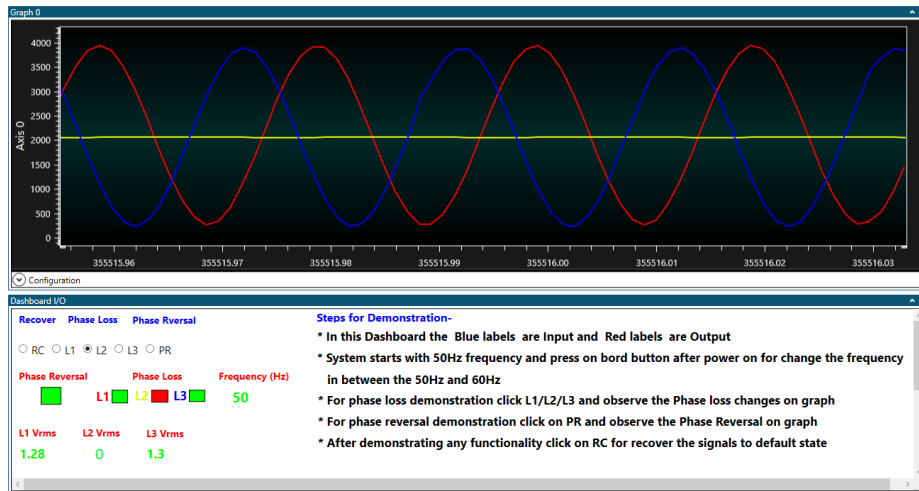
- Connect the hardware and prepare the demonstrator setup, following the instructions provided in the [Demo Setup](#) section.
- Configure the data visualizer, following the instructions provided in the [Data Visualizer Configuration](#) section.
- Click the **Serial Port Control Panel Connect** button.
- Click the **SW0** button on the Curiosity Nano board and select the appropriate frequency for the three-phase signal. The selected frequency is displayed on the custom dashboard window.
- Zoom in or out the graph on the Graph window by clicking the +/- magnifier symbol below the Graph window.
- Insert phase loss fault by clicking the **L1** radio button on dashboard, which disables the L1 signal out of three-phase signal to emulate phase loss input. [Figure 7-15](#) shows the three-phase signal with L1 phase-loss fault and the L1 phase-loss status indicator on the dashboard.

Figure 7-15. Three-Phase Input Signal with L1 Phase Loss



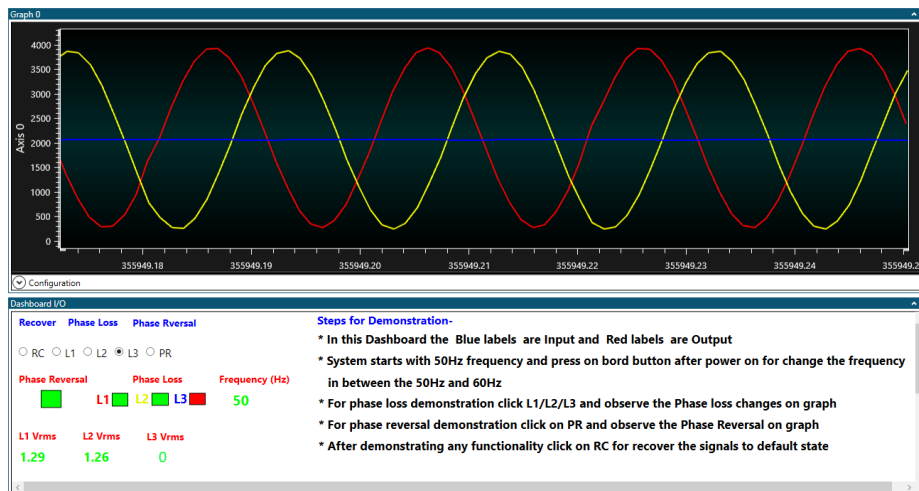
- Insert phase loss fault by clicking the **L2** radio button on dashboard, which disables the L2 signal out of three-phase signal to emulate phase loss input. [Figure 7-16](#) shows the three-phase signal with L2 phase-loss fault and the L2 phase-loss status indicator on the dashboard.

Figure 7-16. Three-Phase Input Signal with L2 Phase Loss



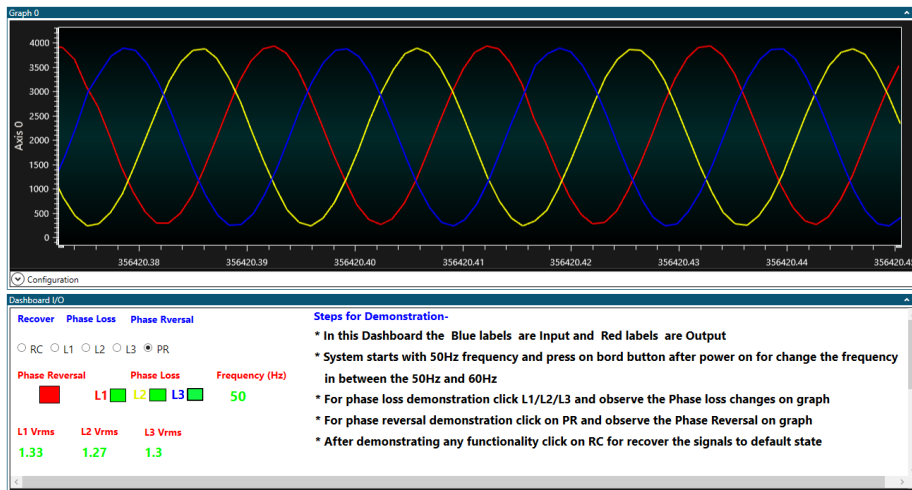
- Insert phase-loss fault by clicking the **L3** radio button on the dashboard, which disables the L3 signal out of three-phase signal to emulate phase-loss input. [Figure 7-17](#) shows the three-phase signal with L3 phase-loss fault and the L3 phase-loss status indicator on the dashboard.

Figure 7-17. Three-Phase Input Signal with L3 Phase Loss



Insert phase-reversal by clicking the **PR** radio button on the dashboard, which changes the sequence of the three-phase signal to L1-L3-L2. [Figure 7-18](#) shows the phase-reversal input and phase-reversal status indication on the dashboard.

Figure 7-18. Three-Phase Input Signal with Phase-Reversal Fault



- Observe the RMS voltage of the three-phase signal on the dashboard

Refer to the [User Inputs](#) section for more information about user input.

## **8. Phase Sequence Detection for Real-Time Applications**

The phase sequence detection solution is appropriate for a wide range of real-time applications, which includes:

- Phase Sequence Meter for multiphase AC systems
- Monitoring relay for lifts
- Escalators
- Pumps
- HVAC Systems
- Irrigation Equipment
- Petrochemical Equipment

## **9. Conclusion**

This implemented emulation model for phase sequence detection of the three-phase AC supply using ZCD, ADC, and timer peripherals of the AVR128DA48 microcontroller. The application note also describes phase reversal, phase loss faults detection, and RMS voltage measurement on the three-phase AC supply using the AVR128DA48 Curiosity Nano development board, Curiosity Nano adapter board, and waveform click boards.

The usage of ZCD enables faster, more precise, and reliable detection of the phase sequence compared to the traditional approach using ADC only, because the ADC related computation time is eliminated. Also, availability of the ZCD peripheral within the microcontroller makes the application more cost-effective and efficient by reducing both board space and BOM (Bill of Materials) cost. The ZCD peripheral together with a rich set of analog and advanced digital peripherals of AVR DA microcontrollers are well-suited for the applications monitoring the phase reversal and single phasing faults of AC power supplies.

## **10. References**

1. [AVR DA Product family page](#)
2. [AVR128DA28/32/48/64 Data sheet](#)
3. [AVR128DA48 Curiosity Nano User Guide](#)
4. [Curiosity Nano Base for Click boards Hardware User Guide](#)
5. [Waveform Click](#)
6. [Data Visualizer User Guide](#)
7. [STK600 AVR Flash MCU Starter Kit User's Guide](#)
8. [STK600 Routing and Socket Boards User Guide](#)

## **11. Appendix**

### **11.1 Application Implementation with AVR128DA64**

The AVR DA family of microcontrollers has up to three instances of ZCD peripheral. In the 64-pin variants of the AVR DA devices, three ZCD instances are available. If all three ZCD instances are used for the phase sequence detection application, the phase loss can also be computed by the ZCD peripheral.

This section explains implementation of phase sequence detection for the three-phase supply using the AVR128DA64 microcontroller. Here in this implementation, TCA0 is used for scheduling the tasks every 500  $\mu$ s; it is configured to generate an interrupt for every 500  $\mu$ s. TCA1 is used for measuring the time in between two ZCD interrupts (i.e. phase-shift in-between the phases of three-phase input); it is configured to generate an overflow interrupt for 20 ms. ADC is used for reproducing the input signal on data visualizer (for demonstration); it is also used for three-phase RMS voltage measurement. The ADC is configured with 2 MHz ADC clock and 126.3 kHz (0.79  $\mu$ s) sampling rate. SPI is used to integrate the waveform click module for generating the three-phase emulated signal; it is configured with Master mode, 6 MHz SPI clock, and data mode 2. The three ZCD instances are used for detecting the zero cross of three-phase input signal; they are configured with an interrupt on the rising edge signal. USART is used for communication between the host computer and application setup; it is configured for a baud rate of 230400.

#### **Firmware Implementation**

The phase sequence detection for the three-phase supply application firmware is split into six sections:

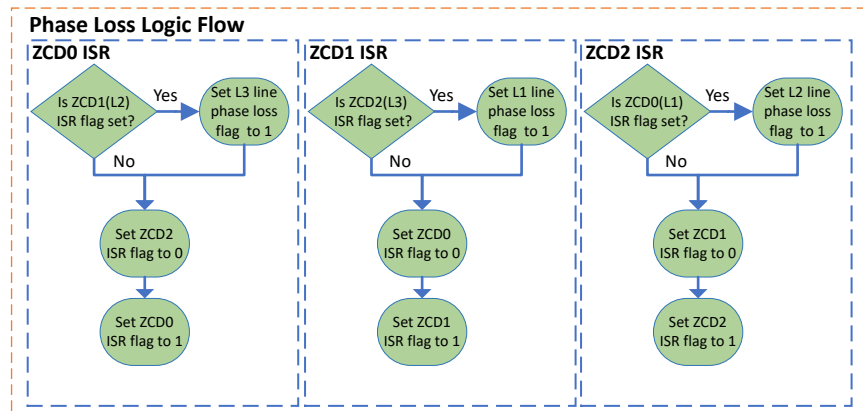
- Initialization
- Phase reversal detection logic
- Phase loss detection logic
- RMS voltage measurement
- Task scheduler
- Emulator functionalities

The phase reversal detection and RMS voltage measurement are implemented as presented in the [Application Firmware Implementation](#) section. The single phasing or phase loss detection functionality is implemented using three ZCD instances, as the AVR128DA64 microcontroller comes with three ZCD instances.

The phases of the three-phase input signals L1, L2, and L3 are connected to three ZCD input, ZCD instances are configured to generate interrupt when the rising input signal voltage crosses the threshold voltage. Here, the 120° phase shift between the three signals can be determined seamlessly, as the three phases of the input signal are connected to the respective ZCD instances. Phase loss can be identified by monitoring the rising edges of the input signal in the expected time.

In each ZCD instance Interrupt Service Routine (ISR), the occurrence of zero-crossing interrupt on all the three ZCD instances is monitored to detect the phase loss event. The application flow diagram of phase loss detection is shown in [Figure 11-1](#).

Figure 11-1. Logic Diagram of Phase Loss Detection Using Three ZCD's



The application firmware of the phase sequence detector demo using AVR128DA64 is available to download from GitHub.



**View Code Example on GitHub (Atmel Studio)**

[Click to browse repository](#)



**View Code Example on GitHub (MPLAB® X)**

[Click to browse repository](#)

### Notes:

1. The firmware is developed with the [STK600 development board](#). The kit offers access to all device pins and several useful hardware functions, such as push buttons, LEDs, and DataFlash to create a complete system for prototyping and testing new designs.
2. The AVR device connects to the STK600 using an innovative routing (STK600-RC064AVR-108) and socket ([STK600-TQFP64-2](#)) card sandwich system, which routes the signals from the device to the appropriate hardware.

## 11.2 Code Configurator

This section provides the steps used to generate the initialization code from the Microchip code configurators START and MCC.

### 11.2.1 Start Configuration

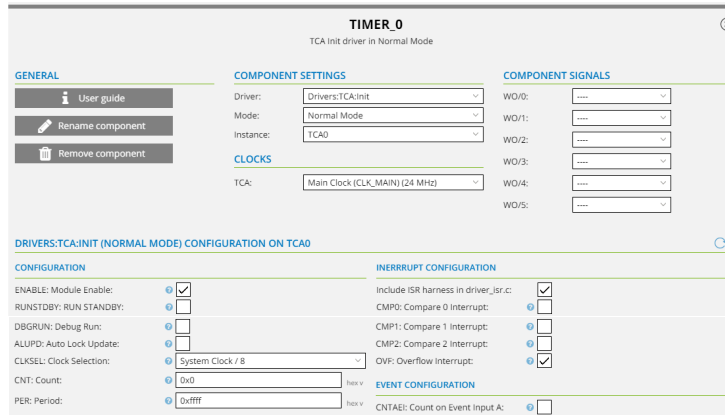
The following peripherals need to be configured for realizing the three-phase phase sequence detection demo:

- Timer/Counter Type A (TCA0)
- Timer/Counter Type A (TCA1)
- USART 1
- SPI 0
- ZDC 0
- ZCD 1
- ADC 0

### 1. Timer 0:

Timer 0 is configured with the system clock divided by 8 frequency and it is used for schedule time for executing the phase sequence detector application functionalities. [Figure 10-2](#) shows the START TIMER 0 configuration window.

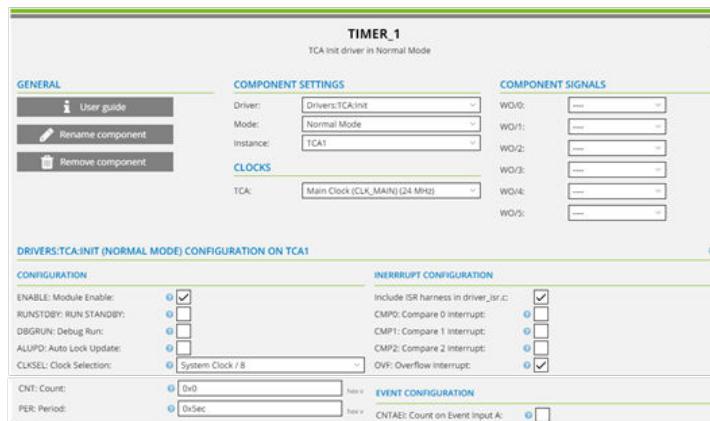
**Figure 11-2. Atmel START Timer 0 Configuration Window**



### 2. Timer 1:

Timer 1 is configured with the system clock divided by 8 frequency and it is used for calculating the phase shift between the two signals out of the three-phase input. [Figure 10-3](#) shows the START TIMER 1.2 configuration window.

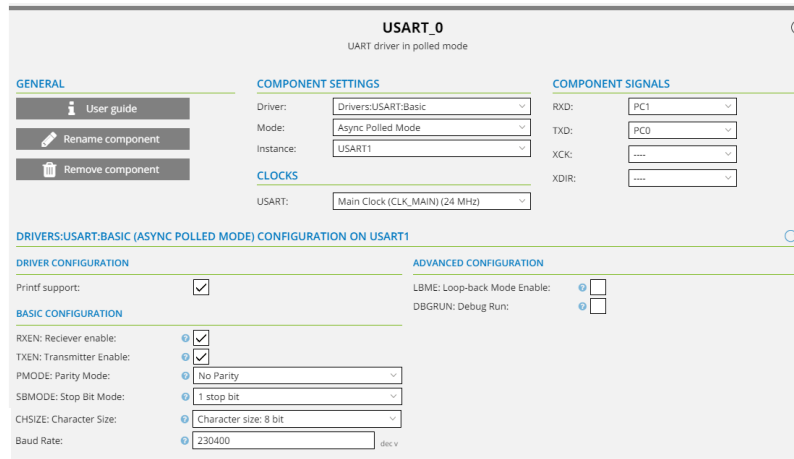
**Figure 11-3. Atmel START Timer 1 Configuration Window**



### 3. USART 1:

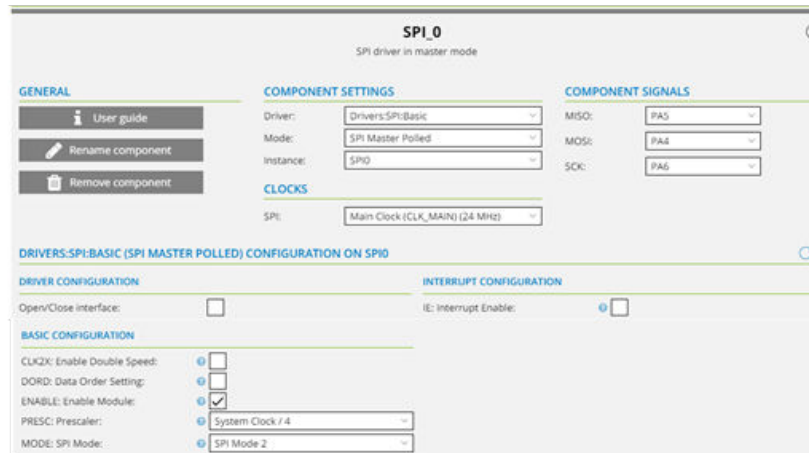
USART is configured with a 230400 baud rate, interrupt enabled, and it is used for communication between the application and Data Visualizer Graphical tool (PC tool). [Figure 10-4](#) shows the START USART configuration window.

**Figure 11-4. Atmel START USART Configuration Window**



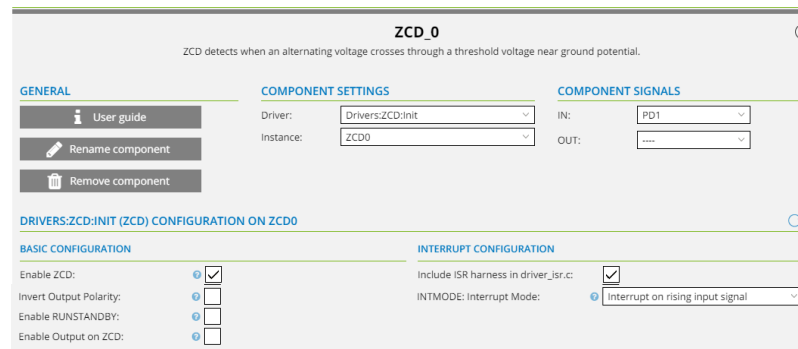
4. **SPI 0:**  
SPI is configured with a 6 Mhz speed, in mode 2 and it is used for interface signal emulator modules. [Figure 10-5](#) shows the START SPI configuration window.

**Figure 11-5. Atmel START SPI Configuration Window**



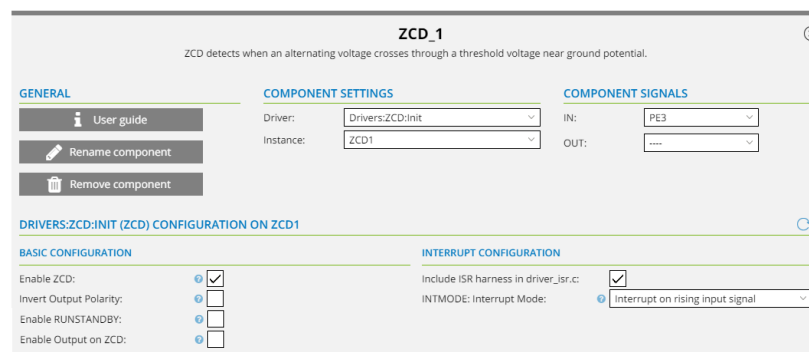
5. **ZCD 0:**  
ZCD 0 is configured with the rising edge detection interrupt enable and it is used for detecting the phase of the three-phase signal. [Figure 10-6](#) shows the START ZCD 0 configuration window.

**Figure 11-6. Atmel START ZCD 0 Configuration Window**



6. **ZCD 1:**  
ZCD 1 is configured with the rising edge detection interrupt enable and it is used for detecting the phase of the three-phase signal. [Figure 10-7](#) shows the START ZCD 1 configuration window.

**Figure 11-7. Atmel START ZCD 1 Configuration Window**



7. **ADC 0:**  
ADC 0 is configured with the Differential mode enabled in Polling mode and it is used for measuring the three-phase signal to reproduce the signal on the data visualizer, calculating the RMS voltage and the single phasing detection. [Figure 10-8](#) shows the START ADC 0 configuration window.

**Figure 11-8. Atmel START ADC 0 Configuration Window**

**ADC\_0**  
MCode basic driver: Peripheral initialization + API to support simple ADC use-case

**GENERAL**

- User guide
- Rename component
- Remove component

**COMPONENT SETTINGS**

Driver: Drivers:ADC:Basic

**CLOCKS**

ADC: Main Clock (CLK\_MAIN) (24 MHz)

**COMPONENT SIGNALS**

AIN/0:	<input type="checkbox"/> PD0 (PORT/P/20)
AIN/1:	<input type="checkbox"/> PD1 (ZCD_0/IN)
AIN/2:	<input type="checkbox"/> PD2 (PORT/P/22)
AIN/3:	<input type="checkbox"/> PD3
AIN/4:	<input checked="" type="checkbox"/> PD4
AIN/5:	<input checked="" type="checkbox"/> PD5
AIN/6:	<input checked="" type="checkbox"/> PD6
AIN/7:	<input type="checkbox"/> PD7 (PORT/P/27)
AIN/8:	<input type="checkbox"/> PE0
AIN/9:	<input type="checkbox"/> PE1 (PORT/P/31)
AIN/10:	<input type="checkbox"/> PE2 (PORT/P/32)
AIN/11:	<input type="checkbox"/> PE3 (ZCD_1/IN)
AIN/16:	<input type="checkbox"/> PF0
AIN/17:	<input type="checkbox"/> PF1
AIN/18:	<input type="checkbox"/> PF2
AIN/19:	<input type="checkbox"/> PF3 (PORT/P/37)
AIN/20:	<input type="checkbox"/> PF4
AIN/21:	<input type="checkbox"/> PF5

**DRIVERS:ADC: BASIC (ADC BASIC) CONFIGURATION ON ADC0**

**ADVANCED CONFIGURATION**

DBG RUN: Debug run:

RUNSTBY: Run standby mode:

RESSEL: ADC Resolution: 12-bit mode

**MODE AND INPUTS CONFIGURATION**

CONVMODE: Differential Mode Conversion: Differential Mode Conversion

**DIFFERENTIAL MODE CONVERSION**

MUXNEG: Analog Channel Selection Bits: Ground

MUXPOS: Analog Channel Selection Bits: ADC input pin 0

**INTERRUPT CONFIGURATION**

RESRDY: Result Ready Interrupt Enable:

**BASIC CONFIGURATION**

ENABLE: ADC Enable:

PRESC: Clock Prescaler: CLK\_PER divided by 4

LEFADJ: Left Adjust Result:

**SAMPLING CONFIGURATION**

Enable:

**8. ZCD 2:**

ZCD 2 is configured with the rising edge detection interrupt enable and it is used for detecting the phase of the three-phase signal (ZCD 2 peripheral is available only in 64 pin AVR DA family MCUs). Figure 10-9 shows the START ZCD 2 configuration window.

**Figure 11-9. Atmel START ZCD 2 Configuration Window (AVR128DA64)**

**ZCD\_2**  
ZCD detects when an alternating voltage crosses through a threshold voltage near ground potential.

**GENERAL**

- User guide
- Rename component
- Remove component

**COMPONENT SETTINGS**

Driver: Drivers:ZCD:Init

Instance: ZCD2

**COMPONENT SIGNALS**

IN: PE7

OUT: ---

**DRIVERS:ZCD:INIT (ZCD) CONFIGURATION ON ZCD2**

**BASIC CONFIGURATION**

Enable ZCD:

Invert Output Polarity:

Enable RUNSTANDBY:

Enable Output on ZCD:

**INTERRUPT CONFIGURATION**

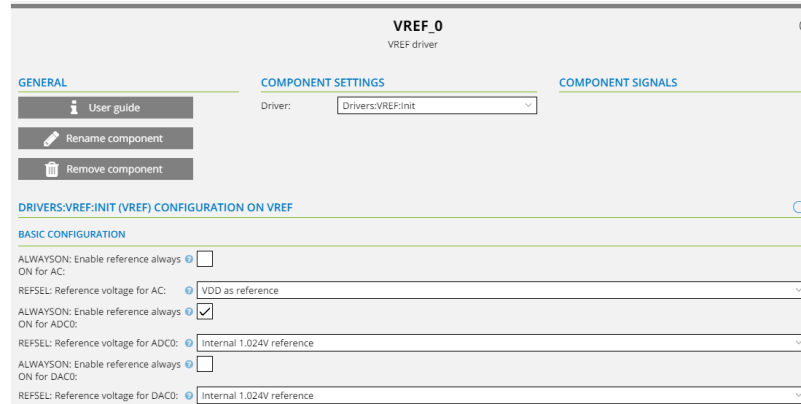
Include ISR harness in driver\_isr.c:

INTMODE: Interrupt Mode: Interrupt on rising input signal

**9. VREF:**

VREF is always configured with the enable reference voltage (VDD) for ADC and it is used for ADC peripheral as a reference voltage. [Figure 10-10](#) shows the START VREF configuration window.

**Figure 11-10. Atmel START VREF Configuration Window**



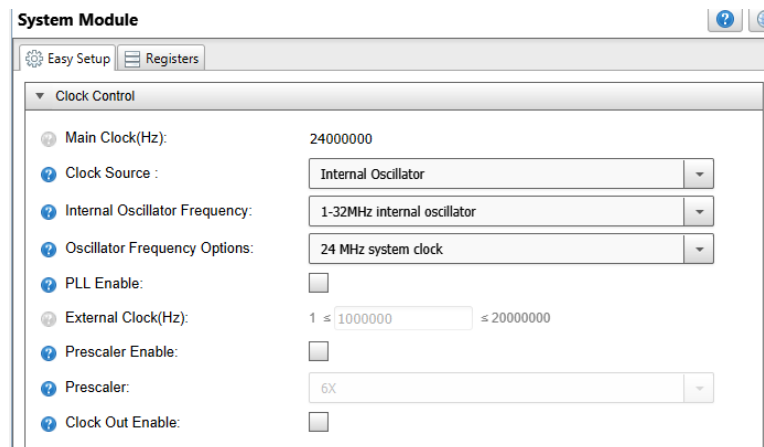
**11.2.2 MCC Configuration**

The three-phase detection application is also generated using the MCC platform, and the microcontroller peripherals and the purpose are discussed in the [START Configuration](#) section.

**System Module:**

[Figure 10-11](#) shows the System configuration window.

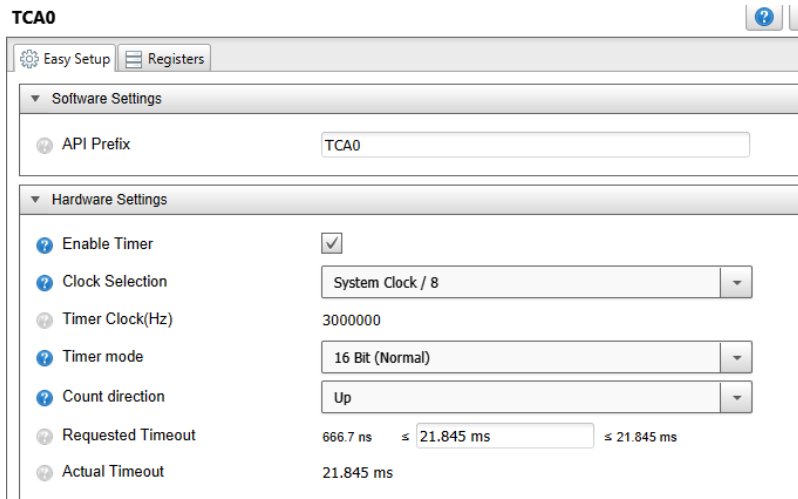
**Figure 11-11. MCC System Configuration Window**



**TIMER 0:**

[Figure 10-12](#) shows the MCC TIMER 0 configuration window.

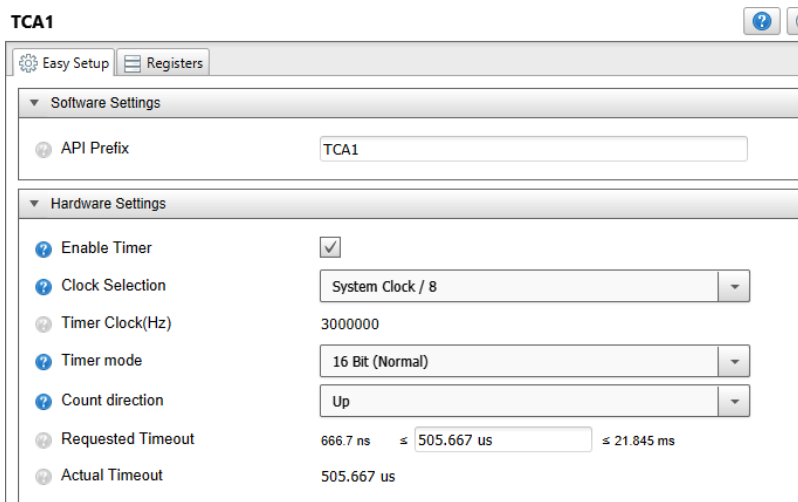
**Figure 11-12. MCC TIMER 0 Configuration Window**



**Timer 1:**

Figure 10-13 shows the MCC TIMER 1 configuration window.

**Figure 11-13. MCC TIMER 1 Configuration Window**



**SPI 0:**

Figure 10-14 shows the MCC SPI 0 configuration window.

**Figure 11-14. MCC SPI 0 Configuration Window**

**SPI0**

Easy Setup | Registers

**Software Settings**

- API Prefix: SPI0
- Interrupt Driven:

**Hardware Settings**

- Enable SPI:
- Modes: Master
- SPI Mode: SPI Mode 2
- Desired Clock Speed (Hz):  $1953 \leq 6000000 \leq 10000000$
- Actual Clock Speed (Hz): 6000000.0

### USART 1:

Figure 10-15 shows the MCC USART 1 configuration window.

**Figure 11-15. MCC USART 1 Configuration Window**

**USART1**

Easy Setup | Registers

**Software Settings**

- API Prefix: USART1
- Interrupt Driven:
- RX Buffer Size (Bytes): 16
- TX Buffer Size (Bytes): 16
- Printf support:

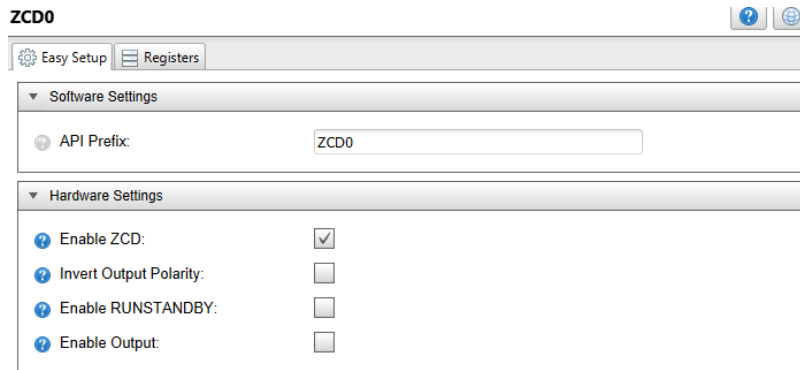
**Hardware Settings**

- Mode: Async Mode
- Baud Rate:  $1 \leq 230400 \leq 1000000$
- Error Percent: -0.161%
- Enable USART Receiver:
- Enable USART Transmitter:
- Parity Mode: No Parity
- Stop Bit Mode: 1 stop bit

### ZCD 0:

Figure 10-16 shows the MCC ZCD 0 configuration window.

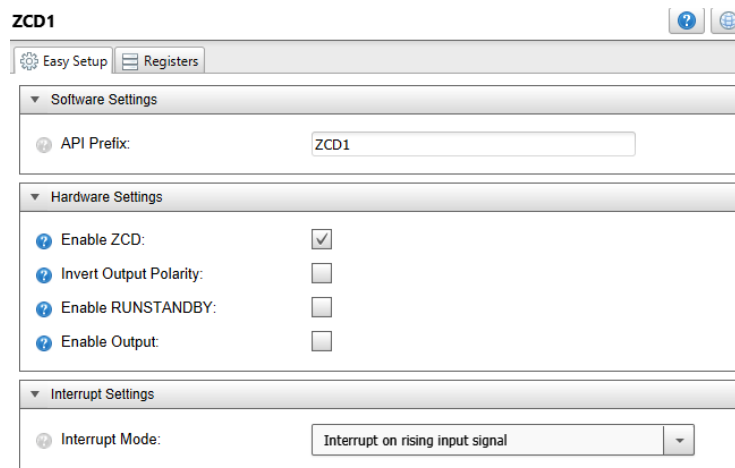
**Figure 11-16. MCC ZCD 0 Configuration Window**



**ZCD 1:**

Figure 10-17 shows the MCC ZCD 1 configuration window.

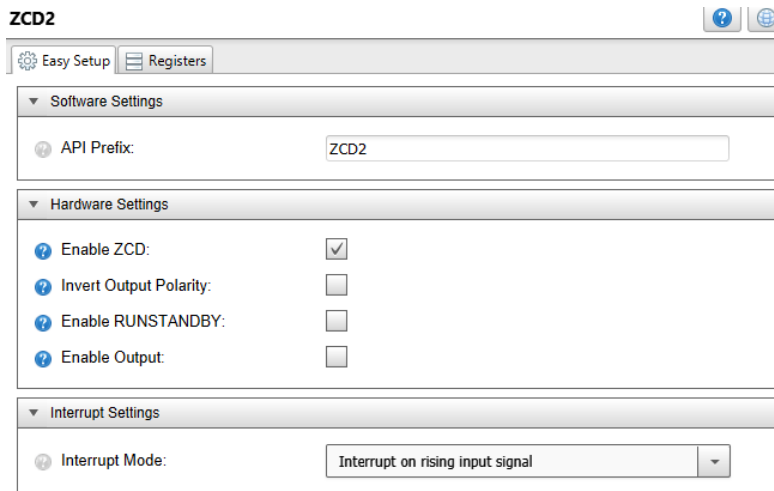
**Figure 11-17. MCC ZCD 1 Configuration Window**



**ZCD 2:**

Figure 10-18 shows the MCC ZCD 2 configuration window.

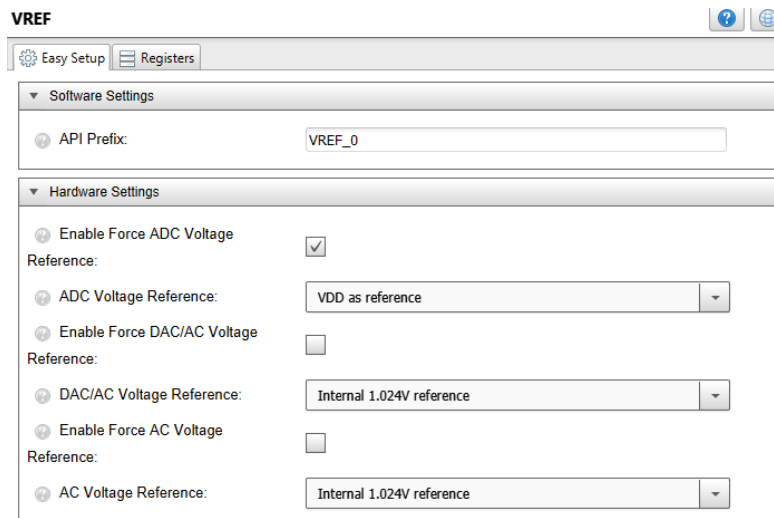
**Figure 11-18. MCC ZCD 2 Configuration Window**



**VREF:**

Figure 10-19 shows the MCC VREF configuration window.

**Figure 11-19. MCC VREF Configuration Window**



**Pin Manager:**

Figure 10-20 shows the MCC Pin Manager configuration window.

Figure 11-20. MCC Pin Manager Configuration Window

Package:	QFN64	Pin No.	62	63	64	1	2	3	4	5	8	9	10	11	12	13	14	15	16	17	18	19	22	23	24	25	26	27	28	29	30	31	32	33	36	37	38	39	40	41	42	43	44	45				
Module	Function	Direction	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1				
ADC0	AINx	input																																														
Pin Module	GPIO	input																																														
	GPIO	output																																														
RSTCTRL	RESET	input																																														
SPI0	MISO	input																																														
	MOSI	output																																														
	SCK	output																																														
TCA0	WO0	output																																														
	WO1	output																																														
	WO2	output																																														
TCA1	WO0	output																																														
	WO1	output																																														
	WO2	output																																														
USART1	RXD	in/out																																														
	TXD	output																																														
ZCD0	OUT	output																																														
	ZCDIN	input																																														
ZCD1	OUT	output																																														
	ZCDIN	input																																														
ZCD2	OUT	output																																														
	ZCDIN	input																																														

### Pin Module:

Figure 10-21 shows the MCC Pin Module configuration window.

Figure 11-21. MCC Pin Module Configuration Window

Pin Name	Module	Function	Custom Name	OUTPUT	START HIGH	INVEN	PULLUPEN	ISC
PA0	TCA0	WO0		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA1	TCA0	WO1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA2	TCA0	WO2		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA4	SPI0	MOSI		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA5	SPI0	MISO		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA6	SPI0	SCK		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA7	Pin Module	GPIO	AD5227_CS1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA8	TCA1	WO0		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA1	TCA1	WO1		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA2	TCA1	WO2		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA4	Pin Module	GPIO	AD5227_CS3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PA5	Pin Module	GPIO	WC_OE2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PC0	USART1	TXD		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PC1	USART1	RXD		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PC7	Pin Module	GPIO	SW	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Interrupt disabl...
PD0	Pin Module	GPIO	WC_OE1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PD1	ZCD0	ZCDIN		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Digital Input Bu...
PD2	Pin Module	GPIO	WC_OE3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PD7	Pin Module	GPIO	AD9833_FS#1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PE1	Pin Module	GPIO	AD9833_FS#2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PE2	Pin Module	GPIO	AD5227_CS2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...
PE3	ZCD1	ZCDIN		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Digital Input Bu...
PE7	ZCD2	ZCDIN		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Digital Input Bu...
PF3	Pin Module	GPIO	AD9833_FS#2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt disabl...

### ADC0:

Figure 10-22 shows the MCC ADC configuration window.

Figure 11-22. MCC ADC 0 Configuration Window

The screenshot shows the 'ADC0' configuration window with two tabs: 'Easy Setup' and 'Registers'. The 'Easy Setup' tab is active and contains two sections: 'Software Settings' and 'Hardware Settings'. Each section has a dropdown arrow on the left. The 'Software Settings' section includes: 'API Prefix' (text field with 'ADC0'), 'Result Selection' (dropdown menu with '12-bit mode'), 'Differential Mode Conversion' (dropdown menu with 'enabled'), and 'Left Adjust Result' (checkbox, unchecked). The 'Hardware Settings' section includes: 'Enable ADC' (checkbox, checked), 'Sampling Frequency(Hz)' (range field with '136363 ≤ 461538 ≤ 461538'), 'ADC Clock(Hz)' (text field with '6000000'), 'Sample Accumulation Number' (dropdown menu with 'No accumulation'), and 'Sample Length (# of ADC Clock)' (range field with '0 ≤ 0 ≤ 31').

Section	Parameter	Value
Software Settings	API Prefix:	ADC0
	Result Selection :	12-bit mode
	Differential Mode Conversion :	enabled
	Left Adjust Result :	<input type="checkbox"/>
Hardware Settings	Enable ADC:	<input checked="" type="checkbox"/>
	Sampling Frequency(Hz):	136363 ≤ 461538 ≤ 461538
	ADC Clock(Hz):	6000000
	Sample Accumulation Number:	No accumulation
	Sample Length (# of ADC Clock) :	0 ≤ 0 ≤ 31

**12. Revision History**

Revision	Date	Description
A	08/2020	Initial document release

---

## The Microchip Website

---

Microchip provides online support via our website at [www.microchip.com/](http://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](http://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](http://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

---

Information contained in this publication is provided for the sole purpose of designing with and using Microchip products. Information regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, COST OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

## Trademarks

---

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Klear, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KlearNet, KlearNet logo, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2020, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-6663-5

## Quality Management System

---

For information regarding Microchip's Quality Management Systems, please visit [www.microchip.com/quality](http://www.microchip.com/quality).

## Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
<p><b>Corporate Office</b> 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Tel: 480-792-7277 Technical Support: <a href="http://www.microchip.com/support">www.microchip.com/support</a> Web Address: <a href="http://www.microchip.com">www.microchip.com</a></p> <p><b>Atlanta</b> Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455</p> <p><b>Austin, TX</b> Tel: 512-257-3370</p> <p><b>Boston</b> Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088</p> <p><b>Chicago</b> Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075</p> <p><b>Dallas</b> Addison, TX Tel: 972-818-7423 Fax: 972-818-2924</p> <p><b>Detroit</b> Novi, MI Tel: 248-848-4000</p> <p><b>Houston, TX</b> Tel: 281-894-5983</p> <p><b>Indianapolis</b> Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380</p> <p><b>Los Angeles</b> Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800</p> <p><b>Raleigh, NC</b> Tel: 919-844-7510</p> <p><b>New York, NY</b> Tel: 631-435-6000</p> <p><b>San Jose, CA</b> Tel: 408-735-9110 Tel: 408-436-4270</p> <p><b>Canada - Toronto</b> Tel: 905-695-1980 Fax: 905-695-2078</p>	<p><b>Australia - Sydney</b> Tel: 61-2-9868-6733</p> <p><b>China - Beijing</b> Tel: 86-10-8569-7000</p> <p><b>China - Chengdu</b> Tel: 86-28-8665-5511</p> <p><b>China - Chongqing</b> Tel: 86-23-8980-9588</p> <p><b>China - Dongguan</b> Tel: 86-769-8702-9880</p> <p><b>China - Guangzhou</b> Tel: 86-20-8755-8029</p> <p><b>China - Hangzhou</b> Tel: 86-571-8792-8115</p> <p><b>China - Hong Kong SAR</b> Tel: 852-2943-5100</p> <p><b>China - Nanjing</b> Tel: 86-25-8473-2460</p> <p><b>China - Qingdao</b> Tel: 86-532-8502-7355</p> <p><b>China - Shanghai</b> Tel: 86-21-3326-8000</p> <p><b>China - Shenyang</b> Tel: 86-24-2334-2829</p> <p><b>China - Shenzhen</b> Tel: 86-755-8864-2200</p> <p><b>China - Suzhou</b> Tel: 86-186-6233-1526</p> <p><b>China - Wuhan</b> Tel: 86-27-5980-5300</p> <p><b>China - Xian</b> Tel: 86-29-8833-7252</p> <p><b>China - Xiamen</b> Tel: 86-592-2388138</p> <p><b>China - Zhuhai</b> Tel: 86-756-3210040</p>	<p><b>India - Bangalore</b> Tel: 91-80-3090-4444</p> <p><b>India - New Delhi</b> Tel: 91-11-4160-8631</p> <p><b>India - Pune</b> Tel: 91-20-4121-0141</p> <p><b>Japan - Osaka</b> Tel: 81-6-6152-7160</p> <p><b>Japan - Tokyo</b> Tel: 81-3-6880-3770</p> <p><b>Korea - Daegu</b> Tel: 82-53-744-4301</p> <p><b>Korea - Seoul</b> Tel: 82-2-554-7200</p> <p><b>Malaysia - Kuala Lumpur</b> Tel: 60-3-7651-7906</p> <p><b>Malaysia - Penang</b> Tel: 60-4-227-8870</p> <p><b>Philippines - Manila</b> Tel: 63-2-634-9065</p> <p><b>Singapore</b> Tel: 65-6334-8870</p> <p><b>Taiwan - Hsin Chu</b> Tel: 886-3-577-8366</p> <p><b>Taiwan - Kaohsiung</b> Tel: 886-7-213-7830</p> <p><b>Taiwan - Taipei</b> Tel: 886-2-2508-8600</p> <p><b>Thailand - Bangkok</b> Tel: 66-2-694-1351</p> <p><b>Vietnam - Ho Chi Minh</b> Tel: 84-28-5448-2100</p>	<p><b>Austria - Wels</b> Tel: 43-7242-2244-39 Fax: 43-7242-2244-393</p> <p><b>Denmark - Copenhagen</b> Tel: 45-4485-5910 Fax: 45-4485-2829</p> <p><b>Finland - Espoo</b> Tel: 358-9-4520-820</p> <p><b>France - Paris</b> Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79</p> <p><b>Germany - Garching</b> Tel: 49-8931-9700</p> <p><b>Germany - Haan</b> Tel: 49-2129-3766400</p> <p><b>Germany - Heilbronn</b> Tel: 49-7131-72400</p> <p><b>Germany - Karlsruhe</b> Tel: 49-721-625370</p> <p><b>Germany - Munich</b> Tel: 49-89-627-144-0 Fax: 49-89-627-144-44</p> <p><b>Germany - Rosenheim</b> Tel: 49-8031-354-560</p> <p><b>Israel - Ra'anana</b> Tel: 972-9-744-7705</p> <p><b>Italy - Milan</b> Tel: 39-0331-742611 Fax: 39-0331-466781</p> <p><b>Italy - Padova</b> Tel: 39-049-7625286</p> <p><b>Netherlands - Drunen</b> Tel: 31-416-690399 Fax: 31-416-690340</p> <p><b>Norway - Trondheim</b> Tel: 47-72884388</p> <p><b>Poland - Warsaw</b> Tel: 48-22-3325737</p> <p><b>Romania - Bucharest</b> Tel: 40-21-407-87-50</p> <p><b>Spain - Madrid</b> Tel: 34-91-708-08-90 Fax: 34-91-708-08-91</p> <p><b>Sweden - Gothenberg</b> Tel: 46-31-704-60-40</p> <p><b>Sweden - Stockholm</b> Tel: 46-8-5090-4654</p> <p><b>UK - Wokingham</b> Tel: 44-118-921-5800 Fax: 44-118-921-5820</p>