
Current Measurement and Battery Monitoring

Features

- Current and Voltage Measurement Theory
- Application Example With ATtiny1627 Curiosity Nano
- Send Measurement Values Over the Curiosity Virtual COM Port
- Visualize Data Using MPLAB® Data Visualizer

Introduction

Author: Marcus Young, Amund Aune, Microchip Technology Inc.

This application note shows how to perform current measurements and monitor a battery in an electronic circuit using a 12-bit Analog-to-Digital Converter (ADC) with a Programmable Gain Amplifier (PGA). Both the current measurements and battery management are, on a theoretical level, explained, followed by a presentation of the advantages of using this ADC.

An application demo using the ATtiny1627 Curiosity Nano is presented. In this demo, multiple voltages are measured, and the corresponding currents are calculated. Then all the metrics are sent over USART to the MPLAB® Data Visualizer, where they are graphed in real-time.

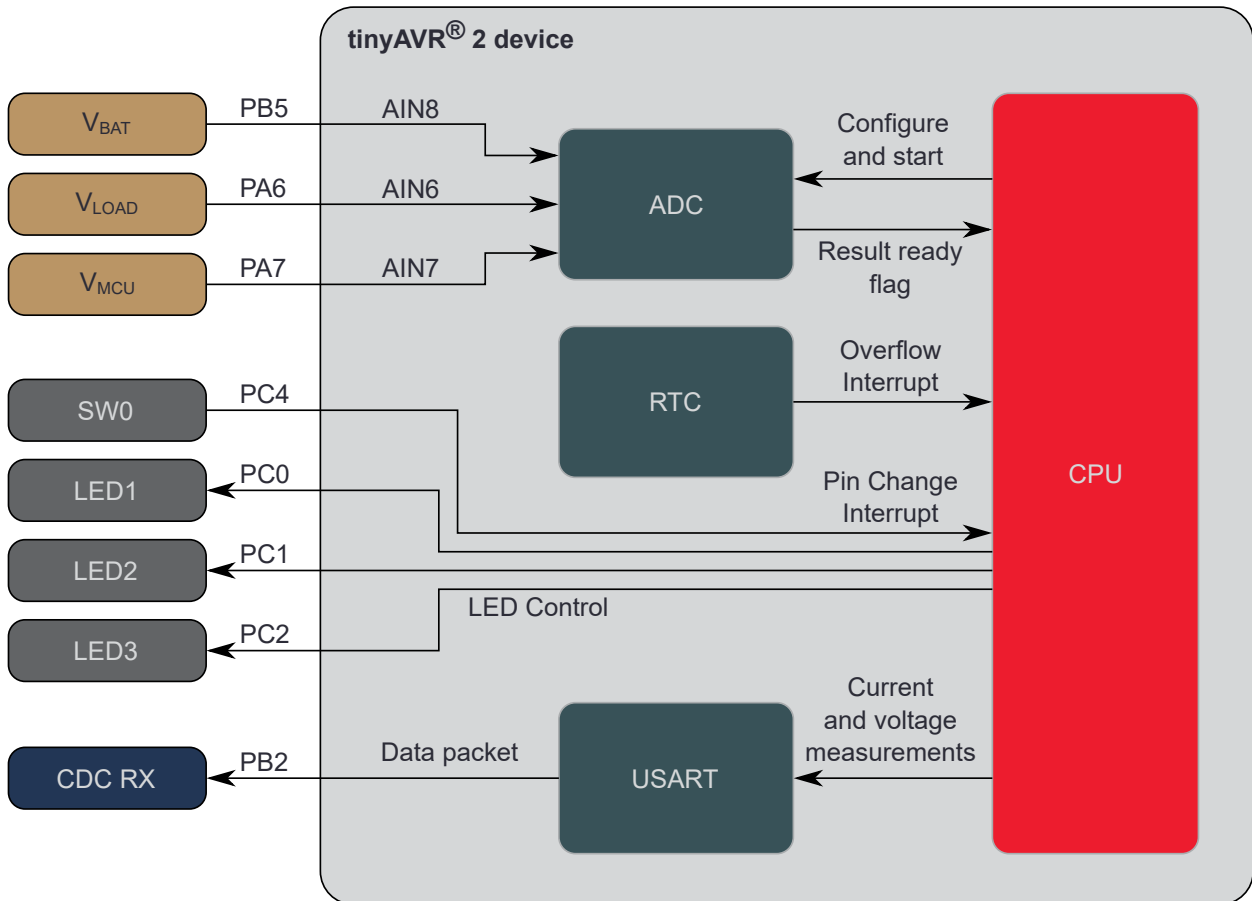
Table of Contents

Features.....	1
Introduction.....	1
1. Block Diagram.....	3
2. Theory of Operation.....	4
2.1. Current Measurement.....	4
2.2. Battery Monitoring.....	5
3. Advantages with the 12-bit ADC and PGA.....	7
4. Demo Operation.....	8
4.1. Hardware Prerequisites.....	9
4.2. Software Prerequisites.....	9
4.3. Running the Example.....	9
5. Source Code Overview.....	10
5.1. Source Code Overview.....	10
5.2. Code Implementation.....	10
6. Plot Graph In MPLAB® Data Visualizer.....	13
7. Get Code Examples from GitHub.....	17
8. Revision History.....	18
Microchip Information.....	19
The Microchip Website.....	19
Product Change Notification Service.....	19
Customer Support.....	19
Microchip Devices Code Protection Feature.....	19
Legal Notice.....	19
Trademarks.....	20
Quality Management System.....	21
Worldwide Sales and Service.....	22

1. Block Diagram

The block diagram below shows an overview of the application example using an ATtiny1627 Curiosity Nano. It shows the connections to the board functions and how they interact with the peripherals and CPU of the tinyAVR® 2 device.

Figure 1-1. Block Diagram



2. Theory of Operation

2.1 Current Measurement

Current cannot be directly measured using an ADC. Since an ADC measures voltage, and according to Ohm's law, the result must be divided by the resistance of the circuit to reflect the current. The current can easily be calculated for a circuit with a constant and known resistance, for example, a simple LED in series with a resistor. However, the circuit resistance is often neither constant nor known in a more complex application.

For systems with variable resistance, a resistor can be added in series with the circuit to be measured. All the current passing through the circuit also passes through the resistor. Therefore, the current may be calculated based on the voltage across it and its resistance value. Usually, this resistor is called a shunt resistor, R_{SHUNT} .

2.1.1 Shunt Resistor Setup

The shunt resistor may be placed on either side of the load to be measured. A high-side measurement is performed if one of the shunt sides connects to the high power rail V_{CC} . If the shunt connects to ground, low-side measurement is used. These techniques both have advantages and disadvantages.

High-side measurements have the advantage that the load is still connected directly to ground. Because of the physical nature of most PCBs, including ground planes close to paths and ground-referenced component casings, a short to ground is more likely to happen than one to a different voltage. If the load shorts to ground with low-side measurements, the short's high current flows around the current measurement shunt resistor, and the short may not be measurable. See [Figure 2-1](#). As shown in [Figure 2-2](#), with high-side measurement, a load short will lead to higher current readings at R_{SHUNT} and can be detected.

Figure 2-1. Short to Ground with Low Side Measurements

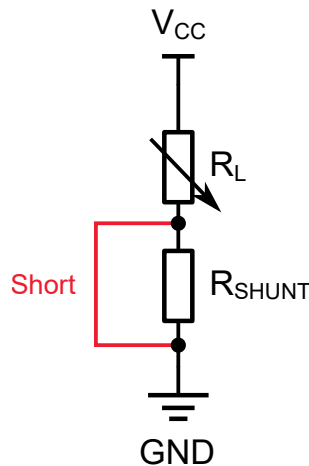
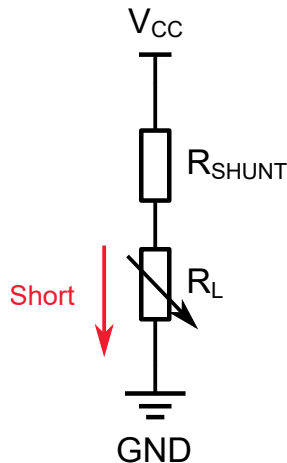


Figure 2-2. Short to Ground with High Side Measurements



Low-side measurements have the advantage of low common mode voltages. As long as the ADC is referenced to the same ground as the circuit it measures, the voltage reading of the load side of the shunt will represent the entire voltage drop. With high-side measurements, on the other hand, both sides of the shunt resistor will be measured to more than 0V, and the voltage difference reflects the current. Running differential conversions lets the ADC sample only the difference between the voltages and utilize the full range of the ADC with 0V corresponding to 0A of current.

2.1.2 Sampling Resolution

The sampling resolution of the current measurement relies on several factors. The ADC resolution is one factor as it sets the ceiling for achievable results. Another factor is the selected measurement range. The shunt can theoretically measure infinite current, but since the ADC supports only a limited number of different results, a more realistic range must be selected. For example, if using a 12-bit ADC, there are 4096 measurement steps. When selecting a maximum current of 4.096A, each step of the ADC would, in the best case, correspond to 1 mA.

Another factor is the matching of the ADC range. The ADC will achieve its lowest result at 0V. With a differential measurement this means that no current is passing through the circuit. However, the highest ADC value is decided by the reference voltage V_{REF} . Make sure that the voltage across the shunt resistor is as close as possible to V_{REF} at I_{MAX} . If it is any lower, all possible samples above this voltage are never reached, and the resolution of the ADC is consequently lowered. A gain stage is often used to amplify the maximum input voltage to as close to V_{REF} as possible.

2.1.3 Shunt Resistance

Since the shunt resistor is in series with the load, choose a resistance as low as possible. The voltage across the shunt resistor is directly proportional to its resistance, and the voltage drop lowers the voltage across the rest of the circuit. A lower voltage supply for the load leads to lower maximum power dissipation P_{MAX} for the load.

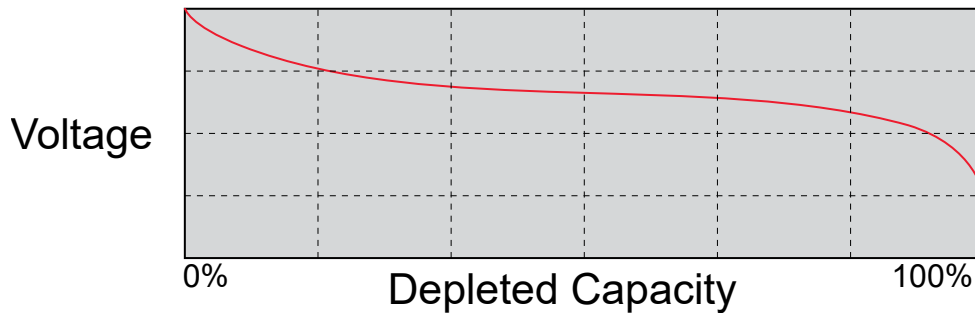
2.2 Battery Monitoring

Keeping track of the battery voltage is battery monitoring in its simplest form. The system either powers off or the application notifies a battery replacement or recharging the battery if its voltage drops below a given value. The lowest acceptable voltage for the battery may be decided by the specified battery voltage range or a component in the system which cannot operate below a certain voltage level.

Depending on the system, the voltage over different components may vary based on more factors than the battery voltage. For example, in systems monitoring the current, the voltage drop across the shunt resistors is proportional to the current that passes through them. In those cases, the voltage may be measured in multiple places or combined with current measurement to ensure that every component is supplied enough voltage.

In most batteries, the discharged capacity is not directly proportional to the voltage. See also the figure below.

Figure 2-3. Example Battery Discharge Curve



When estimating the remaining battery life, this relationship between voltage and capacity must be taken into account. If the ADC reads 3V and the maximum voltage for the battery is 6V, the remaining capacity is not necessarily 50%.

Finding the discharge curve for a specific battery can be done in different ways depending on the desired accuracy. If there is no need for high accuracy, use the discharge curve of a battery of the same type to approximate its behavior. If higher precision is needed, test the battery by applying a constant expected load and monitoring the voltage as it discharges to produce the discharge curve for the specific battery used.

Here, we discuss two methods for monitoring the battery state-of-charge (SoC).

The first method is to use the voltage to estimate SoC. This method would use the measured voltage to access a lookup table (LUT) to get the SoC. This method is simple but will give a poor estimation of the SoC if the mid-section of the discharge curve is flat.

The second method uses voltage monitoring in combination with current monitoring. A widely used method for current monitoring is Coulomb counting, which involves integrating over the current drawn to get the SoC. You can achieve a high accuracy regarding the SoC when using voltage monitoring for the discharge curve's upper, lower, and non-flat parts and current monitoring for the flat mid-section of the discharge curve. However, this method is more complicated to implement than voltage monitoring.

Battery properties change with usage and over time and will therefore need periodical re-calibration. Different capacity estimation methods might require various calibration methods. Running the battery through a complete charge cycle (depleted to fully charged) and then a discharge cycle (fully charged to depleted) is a simple calibration method. Let the battery rest and cool down after each cycle to achieve a more accurate calibration. By doing this, you get new values for updating the discharge curve.

3. Advantages with the 12-bit ADC and PGA

The 12-bit ADC is, in many ways, well suited for current measurement. Its 12-bit resolution gives a good starting point for achieving high resolution for the current measurements. Furthermore, the PGA enables the amplification of the voltage measured over the shunt resistor without the need for external components. The PGA can amplify the voltage up to 16x.

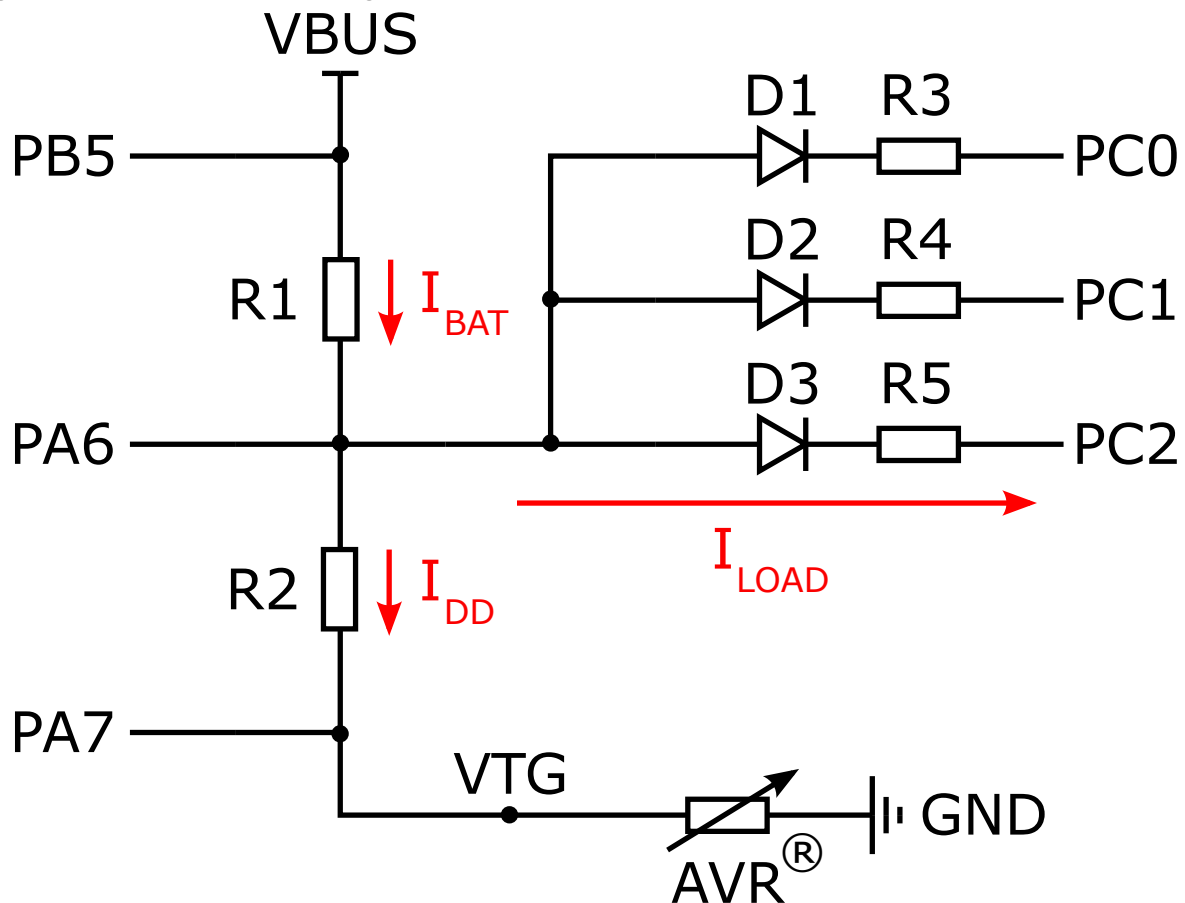
The ADC implements differential measurement to be sure that only the resistor's voltage drop needs to be measured, whether the shunt resistor is connected on the high or the low side. To perform high-side or low-side measurements, the ADC needs to be able to measure voltages above V_{DD} and below V_{SS} , of which this ADC is capable. The differential result will be zero with no current flowing and increase proportionally with the current.

4. Demo Operation

In this demo, we use the ATtiny1627 Curiosity Nano and a breadboard circuit to show how to measure circuit currents and monitor a battery using the 12-bit ADC with PGA. To test the circuit, the battery voltage, V_{BAT} , is replaced with V_{BUS} supplied by the micro-USB cable connected to the Curiosity Nano. Furthermore, disconnect the on-board debugger from the on-board target voltage regulator to test this concept using a supply voltage from the micro-USB cable and shorting the VOFF pin on the Curiosity Nano to ground. Then you can supply the Curiosity Nano with target voltage on the VTG pin. The ATtiny1627 Curiosity Nano Hardware User Guide provides a detailed description.

The breadboard circuit consists of two shunt resistors, connections to pins on the Curiosity Nano, and a variable load consisting of three LEDs and resistors. The shunt resistors are used to calculate different currents. Figure 4-1 shows a diagram of the breadboard circuit. In this example, $R1 = 1\Omega$ and $R2 = 1.5\Omega$. $R3$, $R4$ and $R5$ are chosen based on the current flowing through the LEDs and wanted light intensity. If you decide to change the values for $R1$ or $R2$, make sure to change the source code correspondingly.

Figure 4-1. Breadboard Circuit Diagram



Shunt resistor $R1$ is placed in series with the battery voltage and can therefore be used to measure the total current drawn from the battery. Shunt resistor $R2$ is put in series with V_{DD} and can be used to find the current drawn by the tinyAVR[®] 2 device. By subtracting the current drawn by the microcontroller from the total current drawn from the battery, the current drawn by the variable load is found.

The equations below show how to calculate the total current drawn from the supply (I_{BAT}), the current drawn by the AVR[®] device (I_{DD}), the current drawn by the load (I_{LOAD}), and the battery voltage (V_{BAT}).

Equation 4-1. I_{BAT}

$$I_{BAT} = \frac{V_{R1}}{R1}$$

Equation 4-2. I_{DD}

$$I_{DD} = \frac{V_{R2}}{R2}$$

Equation 4-3. I_{LOAD}

$$I_{LOAD} = I_{BAT} - I_{DD}$$

Equation 4-4. V_{BAT}

$$V_{BAT} = V_{DD} + V_{R1R2}$$

In addition to measuring the current consumption of the different components of the circuit, voltage measurements must monitor the battery. V_{DD} can be calculated by measuring the internal $V_{DD}/10$ ADC channel and multiplying the result by 10. Then the voltage across both shunt resistors can be measured and added to the V_{DD} result to find V_{BAT} . Recharge the battery if V_{BAT} or V_{DD} is too low for the application to operate correctly.

After performing the necessary measurements, they are converted from ADC results to real-world numbers and sent through USART as floating-point values. MPLAB® Data Visualizer receives these values and graphs them in real time. Three LEDs are connected to the load circuit and controlled by the user by clicking the button on the ATtiny1627 Curiosity Nano board. These are used to observe a change in voltage and current as the load resistance is changed. The LED on the Curiosity Nano board will switch on if all the load circuit LEDs are off and switch off if at least one is on.

4.1 Hardware Prerequisites

- ATtiny1627 Curiosity Nano
 - www.microchip.com/en-us/development-tool/DM080104
- Three LEDs, five resistors, wires and a micro-USB cable

4.2 Software Prerequisites

- MPLAB® X IDE (Version 6.00 or newer)
- MPLAB® XC8 Compiler (Version 2.40 or newer)
- MPLAB® Code Configurator (Version 5.1.17 or newer)
- MPLAB® Data Visualizer (Version 1.3.1160 or newer)
- MPLAB® Melody Core (Version 2.2.37 or newer)
- tinyAVR® Device Family Pack (Version 3.0.151 or newer)

4.3 Running the Example

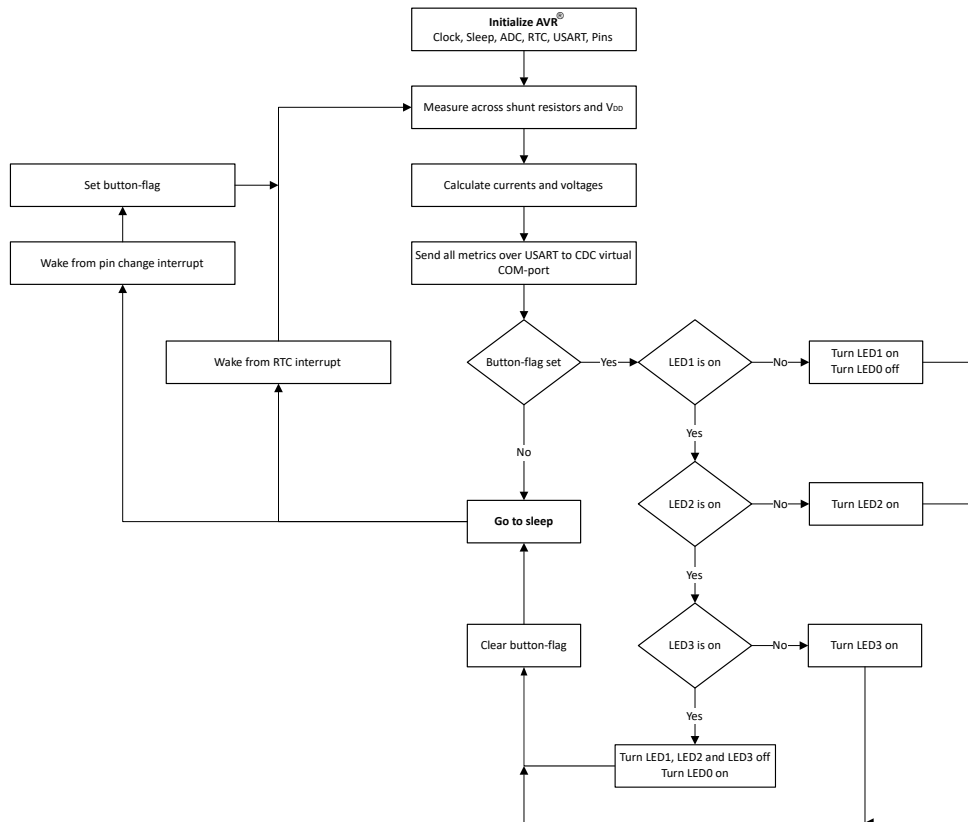
- Download the application as explained in section 7. [Get Code Examples from GitHub](#) and program it to the ATtiny1627 Curiosity Nano
- Connect the LEDs and resistors as shown in [Figure 4-1](#)
- Connect a micro-USB cable between a computer and the Curiosity Nano
- Open MPLAB® X IDE or the standalone MPLAB® Data Visualizer and configure the plotting as explained in section 6. [Plot Graph In MPLAB Data Visualizer](#)
- Press the button on the Curiosity Nano board to change the state of the LEDs to see the current and voltage plots change

5. Source Code Overview

5.1 Source Code Overview

The following flowchart shows the high-level code execution.

Figure 5-1. Code Flow Chart



5.2 Code Implementation

ADC Initialization

The peripherals used in this demo are configured in MPLAB® Code Configurator (MCC). Since the ADC is the peripheral in focus, only the ADC initialization will be described. The initial configuration of the ADC is listed below.

- Burst Accumulation mode
- 256 samples accumulated
- V_{REF} : 1.024V
- Sample duration: 65 CLK_ADC cycles
- Run Standby mode

Note: MCC also configures other settings for the ADC than shown in this list. These settings are reconfigured by the application code before the ADC is started, and are therefore excluded.

Shunt Resistor Measurements

The function below runs when measuring the shunt resistors. The function is called with the ADC channel configurations as arguments.

```
#define OFFSET          0
#define REFERENCE      1.024
#define SAMPLES        256

float adc_diff_16x(ADC_MUXPOS_t muxpos_config, ADC_MUXNEG_t muxneg_config)
{
    /* 16x gain */
    ADC0.PGACTRL = ADC_GAIN_16X_gc | ADC_PGAEN_bm;

    /* Set positive pin */
    ADC0.MUXPOS = ADC_VIA_PGA_gc | muxpos_config;

    /* Enable differential mode and set trigger to MUXNEG write */
    ADC0.COMMAND |= ADC_DIFF_bm | ADC_START_MUXNEG_WRITE_gc;

    /* Set negative pin. This write also starts the burst conversions. */
    ADC0.MUXNEG = ADC_VIA_PGA_gc | muxneg_config;

    /* Wait for result to be ready */
    while(!(ADC0.INTFLAGS & ADC_RESRDY_bm));

    /* Return averaged ADC result */
    /* Amplified and accumulated result divided by the gain value (16), resolution (+/- 2048)
    and number of samples to reflect measured voltage drop */
    return ((int32_t) ADC0.RESULT) + OFFSET) * ((REFERENCE / 16.0) / 2048.0) / SAMPLES);
}
```

The PGA is activated and set to 16x gain, the Differential mode is enabled, the channels are set, and the burst conversions are started. The returned value is the measured voltage drop across the shunt resistance, calculated from the raw accumulated ADC result.

V_{DD} Measurements

The function below runs when measuring V_{DD}. The function is called with the VDD/10 channel as an argument.

```
#define OFFSET          0
#define REFERENCE      1.024
#define SAMPLES        256

float adc_single_2x(ADC_MUXPOS_t muxpos_config)
{
    /* 2x gain */
    ADC0.PGACTRL = ADC_GAIN_2X_gc | ADC_PGAEN_bm;

    /* Set ADC channel */
    ADC0.MUXPOS = ADC_VIA_PGA_gc | muxpos_config;

    /* Enable single ended mode and set trigger to start immediate */
    ADC0.COMMAND = ADC_MODE_BURST_gc | ADC_START_IMMEDIATE_gc;

    /* Wait for result to be ready */
    while(!(ADC0.INTFLAGS & ADC_RESRDY_bm));

    /* Return averaged ADC result */
    /* Amplified and accumulated result divided by the gain value (2), resolution (4096) and
    number of samples to reflect measured voltage drop */
    /* Since the measured voltage is VDD/10, the result is also multiplied by 10 */
    return ((int32_t) ADC0.RESULT) + OFFSET) * 10 * ((REFERENCE / 2.0) / 4096.0) / SAMPLES);
}
```

The PGA is in this function activated but set to only 2x gain because the expected result is higher and will need less amplification to be measured accurately. Like in the previous function, the returned value is the measured voltage calculated from the ADC result.

Use the following calculation to convert the ADC measurement result to a voltage.

Equation 5-1. ADC Voltage Calculation

$$V = \frac{\text{ADCn.RESULT} + \text{OFFSET}}{\text{SAMPNUM}} \times \frac{V_{\text{REF}}}{\text{GAIN} \times \text{RESOLUTION}}$$

The first part of the equation divides the accumulated result with the average from all samples. In an ideal ADC, the first code transition would happen when the input voltage equals 0.5 LSB, resulting in an offset equal to 0, which is not always the case in an actual ADC. This offset holds the deviation value between the ideal first code transition and the first code transition. There exist several methods to calibrate the ADC regarding the offset. Then the averaged ADC result is converted to a voltage by multiplying with a factor of voltage per LSB.

6. Plot Graph In MPLAB® Data Visualizer

MPLAB *Data Visualizer* is a program for processing and visualizing data from a running embedded target and can be accessed as a plugin or a stand-alone program. This application note describes the configuration of the Data Visualizer to graph ADC acquisitions received over USART. The configuration is done using a saved workspace, and the basics of displaying the data are explained. Click the *Documentation* button in MPLAB *Data Visualizer* for a detailed guide on how to set up your workspace.



To do: Configure MPLAB *Data Visualizer* to graph received ADC samples.

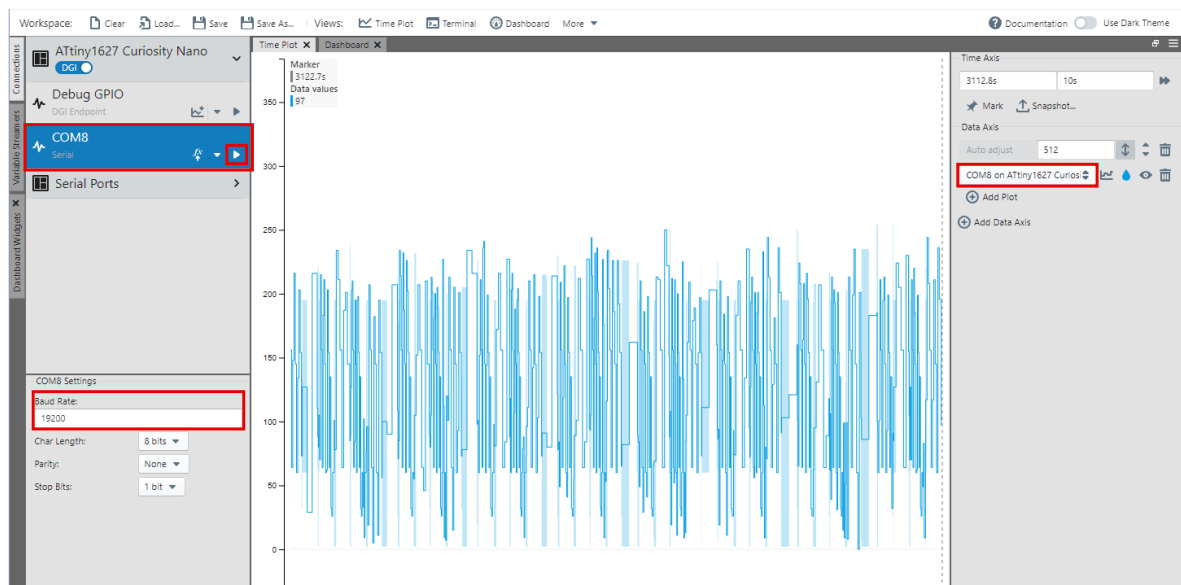
1. MPLAB® X IDE has an internal Data Visualizer. Click on the DV button at the top to open it. If this button is not there, go into Tools and Plugins to install the Data Visualizer.

Figure 6-1. Opening the Data Visualizer in MPLAB® X IDE



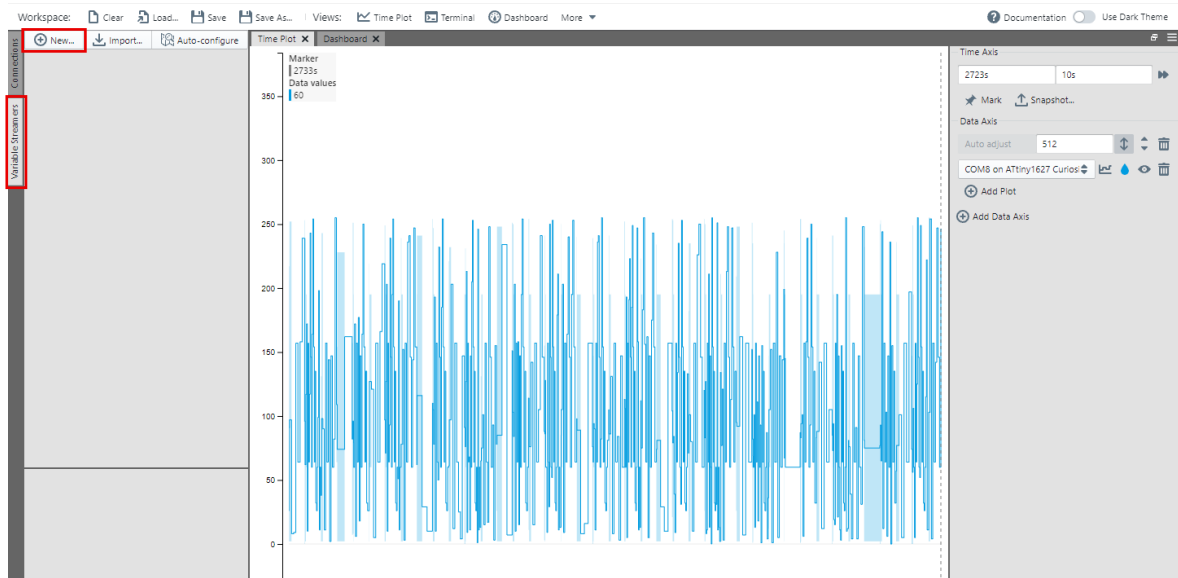
2. Open the program and plug in an already flashed device. Make sure the COM-port used for USART communication is not already in use. The start screen will look similar to the one in Figure 6-2, apart from the plotted graph.

Figure 6-2. MPLAB® Data Visualizer Starting Page



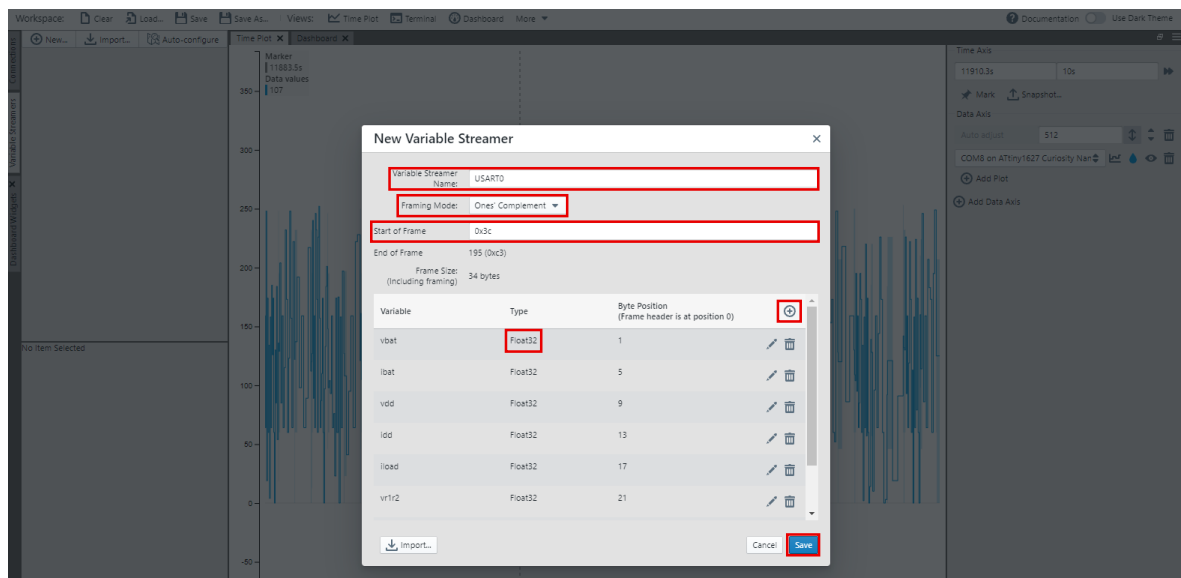
- a. Highlight the correct COM-port by clicking it.
 - b. Set Baud rate to 19200.
 - c. On the right-hand side, select ATtiny1627 from the source drop-down menu.
 - d. Click on the play button.
 - e. If the Data Visualizer plots a graph similar to the one in Figure 6-2, the USART configuration is set up correctly.
 - f. Click the stop button.
3. Find and open the Variable Streamer configuration panel to graph the streamed data output of the device.

Figure 6-3. Open the Variable Streamer



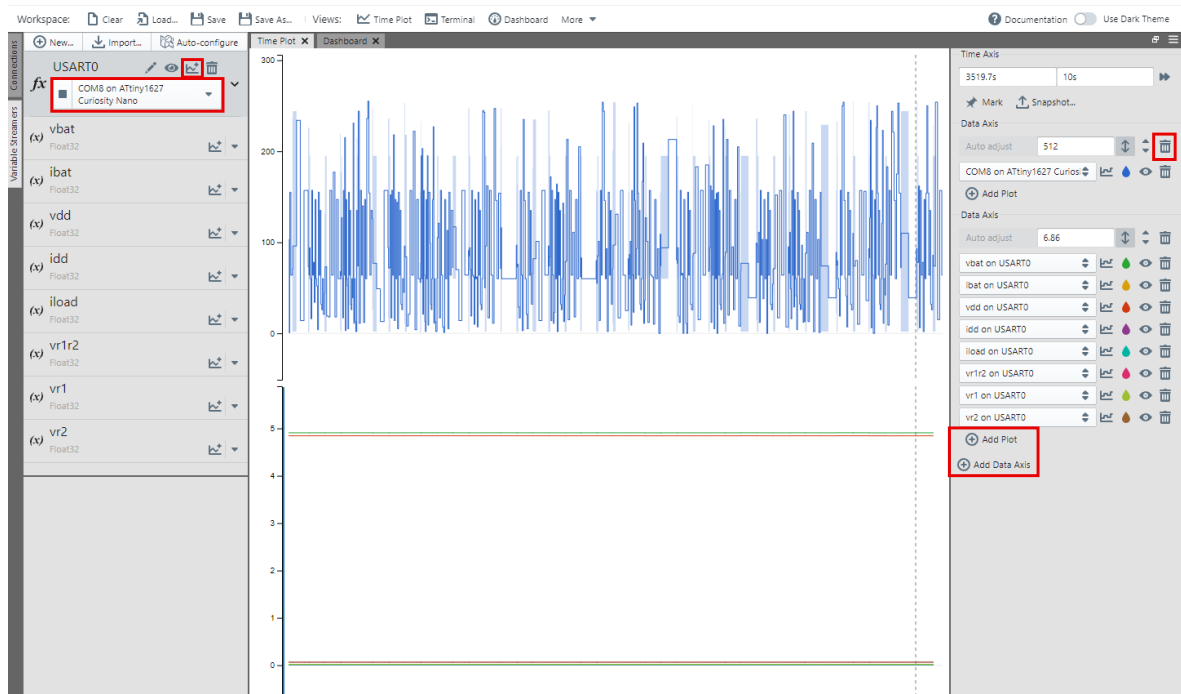
- a. Click the Variable Streamers panel on the left-hand side.
- b. Click on New to open the Variable Streamer configuration panel.
4. Configure the Variable Streamer to decode the streamed data output according to the Stream Format.

Figure 6-4. Configure Variable Streamer



- a. Set a Variable Streamer Name.
- b. For Framing Mode, select Ones' Complement from the drop-down menu.
- c. Set the Start of Frame byte to 0x3c.
- d. Add variables by clicking the plus sign. Remember to add them in the correct order, as shown in [Figure 6-5](#).
- e. For Type, set all variables to Float32.
- f. Click Save in the bottom right when all variables have been added.

Figure 6-5. Plot Variables

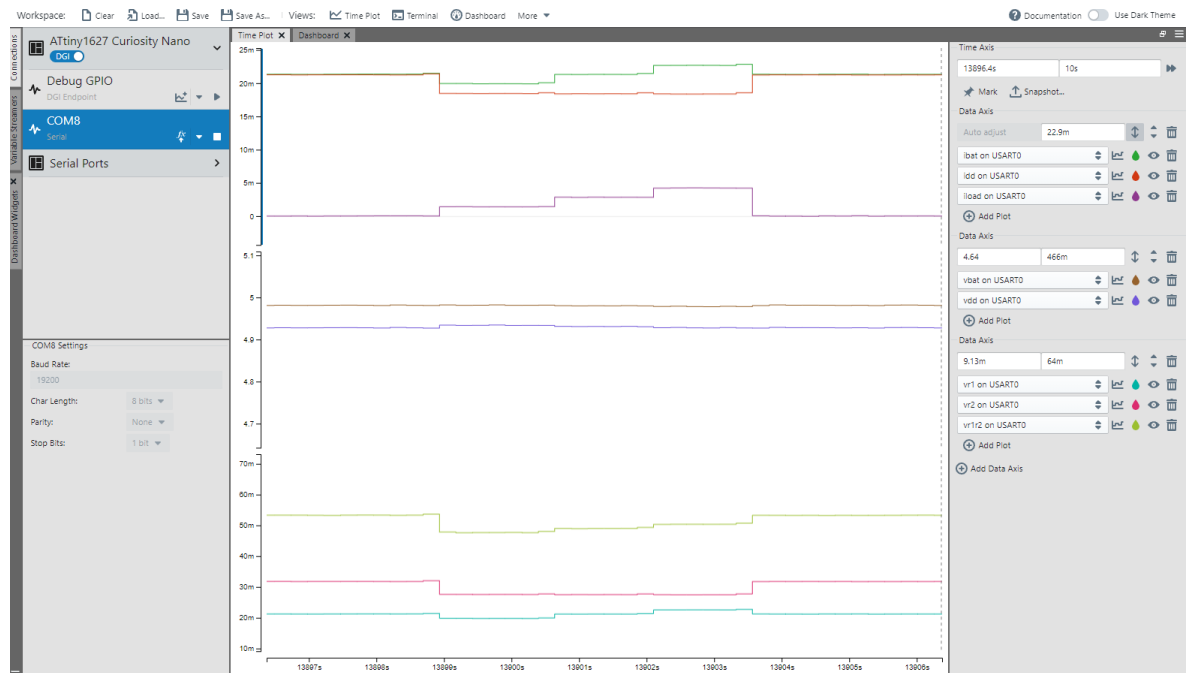


- a. Select the correct COM-port from the source drop-down menu.
 - b. Click the graph icon at the top to plot all variables.
 - c. You can now delete the data axis containing the non-arranged data by clicking the trash can on the right-hand side.
 - d. Add Plot allows you to add a new source to plot in a given data axis set.
 - e. Add Data Axis allows you to add a new set of axes to plot several sources.
 - f. Figure 6-6 shows an example of how to group the different variables in different axes to give a better visual representation of the values.
5. Click *Show Live Data* to start plotting live data from the device. If set to auto-adjust, the axes will scale according to the input from USART. Figure 6-6 shows an example of the plotted result when pushing the button on the Curiosity Nano, which cycles through the four different states in which the load on the breadboard circuit can be.



Info: By clicking on “Save Workspace”, Data Visualizer generates a file that can be used for an agile loading of the workspace by clicking “Load Workspace” and choosing the generated file.

Figure 6-6. Plot Configuration Completed



Result: The MPLAB *Data Visualizer* has been configured to graph data received from the nEDBG USART serial gateway.

7. Get Code Examples from GitHub

The code example is available through GitHub, a web-based server that provides the application code through a Graphical User Interface (GUI). The code example can be opened in MPLAB X.

The GitHub webpage: [GitHub](#).

Code Examples



View Code Examples on GitHub

Click to browse repositories

Download the code as a `.zip` file from the example page on GitHub by clicking the **download** button, or you can clone the repository by clicking the **Clone** button.

8. Revision History

Doc. Rev.	Date	Comments
A	12/2022	Initial document release

Microchip Information

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 products:

- 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, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- 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. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications 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. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

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

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePicta, TimeProvider, TrueTime, 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, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, 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, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQL, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, Trusted Time, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, 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.

© 2022, Microchip Technology Incorporated and its subsidiaries. All Rights Reserved.

ISBN: 978-1-6683-1484-5

Quality Management System

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

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
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	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	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	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