

# Bridging the



## PIC32MM

Lowest-Power and Most  
Cost-Effective Family  
of PIC32 MCUs

## Gap

**7** Taking on  
the World

**17** Jump Start Your  
Internet of Things  
Designs

**20** Solving the  
Security  
Conundrum

## COVER STORY

### 4 Bridging the Gap

Low-Power, Cost-Effective PIC32MM Family of Microcontrollers Offers Compelling 32-bit Solution for Applications with Budget, Power and Size Constraints

## NEW PRODUCTS

### 7 Taking on the World

Next-Generation Bluetooth® Low Energy Solutions Deliver Easy-to-Use Interface and Embedded Scripting Capability

### 9 Now That Was Easy

Add Wi-Fi® Connectivity to Your Design with the ATWINC1500 Wi-Fi Network Controller

### 11 Music Minus Wires

Dual-Mode Bluetooth Audio Devices Deliver Superior Sound Quality for High-End Headsets, Speakers and Sound Bars

## NEW TOOLS

### 13 Get Ready to Innovate

Delivering Over Four Times More Battery Life and Faster Time to Market, New Platform Includes MCU, Bluetooth Low Energy and Sensor Technology

### 14 Weighing In

Add Intelligence and Bluetooth Connectivity to Digital Weight Scales with Microchip's Flexible and Cost-Effective Solutions

## DESIGN CORNER

**15** Jump Start Your Internet of Things Designs

**17** Unifying the Internet of Things

**20** Solving the Security Conundrum

**23** Rapid Prototyping, Seamless Integration

**24** Custom Control

**25** What Is the True Cost?

**27** Simplifying Clock Generation

**30** Smarter Programming

**33** Measuring Time

## FEATURED VIDEO

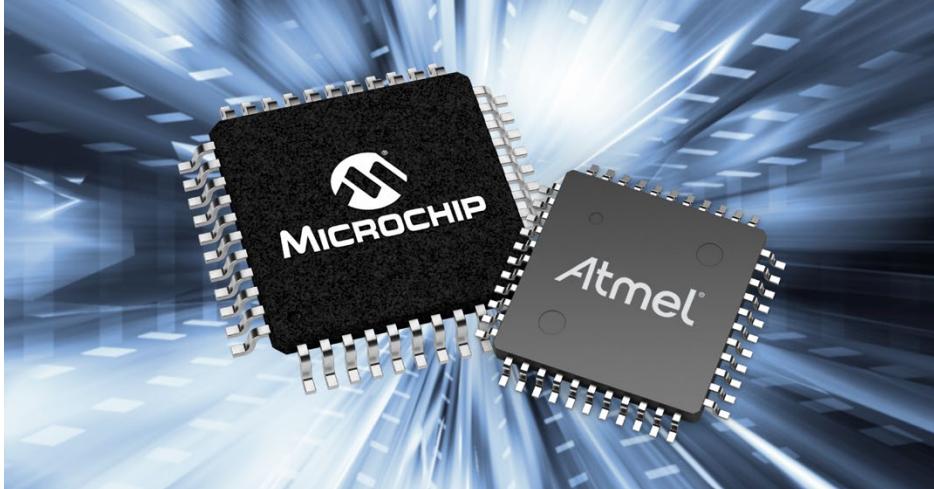


Remote Temperature Reading Across Vacant Lot with MCP9902; 1°C Accurate Down to -40°C

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, KEELOQ, KEELOQ logo, MOST, MPLAB, mTouch, PIC, PICmicro, PICSTART, PIC32 logo, Quiet-WIRE, rfPIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MTP, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries. Atmel and AVR are registered trademarks of Atmel, a wholly owned subsidiary of Microchip Technology Inc.

Analog-for-the-Digital Age, Application Maestro, BodyCom, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FarSense, HI-TIDE, In Circuit Serial Programming, ICSP, Mindi, MIWI, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, Omniscent Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, SQI, Serial Quad I/O, Total Endurance, TSHARC, UniWinDriver, WiperLock, ZENA and Z-Scale 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. GestIC and ULPP are registered trademarks of Microchip Technology Germany II GmbH & Co. & KG, a subsidiary of Microchip Technology Inc., in other countries. The LoRa name and associated logo are registered trademarks of Semtech Corporation or its subsidiaries. USB Type-C™ is a trademark of USB Implementers Forum All other trademarks mentioned herein are property of their respective companies. © 2016, Microchip Technology Incorporated, All Rights Reserved.



## EDITOR'S NOTE

# United to Serve You

Both Microchip and Atmel® have long histories of offering quality, leading-edge and award-winning development tools. As we continue to move towards joining the forces of both companies, we realize that the right tools are imperative for improving your productivity and time to market. Therefore, we intend to support the Microchip and Atmel suites of tools going forward.

Atmel Studio IDE and MPLAB® X IDE are a vital part of their respective ecosystems, each offering unique benefits to their users. For now, Atmel Studio and MPLAB X IDE will both continue to be used for their related device support and there will be no change in support, delivery mechanisms or release schedules. In addition, MPLAB XC and Atmel GCC compilers remain a staple of project development. Professionals, students and makers depend on them for their daily operations. The IAR and Keil partnerships are an important part of the Atmel “classic” story, particularly for AVR® and the SMART line of ARM® processor-based microcontrollers. We intend to continue to foster these relationships.

Having reliable device debuggers and programmers is a critical part of the design process. The Atmel-ICE and the MPLAB In-circuit Debuggers will continue to provide you with that capability.

We also recognize the significant part our third-party partners have played in the success of Microchip and Atmel, and we also intend to continue maintaining and cultivating these relationships going forward. Given time, we will seek synergies that both make sense for the entire client base of Atmel and Microchip and respect the needs and interests of all customers.

Our development tools team, in conjunction with microchipDIRECT, is working to bring top-selling Atmel classic development tool boards into Microchip’s warehouses and make them available for sale through Microchip’s distribution channels. We look forward to continuing to serve you with a diverse and quality selection of development tools.

As always, we would be happy to get your feedback on MicroSolutions. Feel free to email us at [MSFeedback@microchip.com](mailto:MSFeedback@microchip.com).

---

Microchip Technology Inc.  
2355 W. Chandler Blvd. | Chandler, AZ 85224 | [www.microchip.com](http://www.microchip.com)

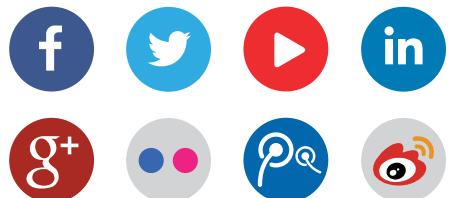


## Don't Miss the Next Issue of MicroSolutions

Published six times a year, MicroSolutions is a valuable resource for product and technology news about Microchip’s innovative solutions. Subscribe today to receive email notifications when each new issue of MicroSolutions is ready. Use the link below:

[CLICK TO SUBSCRIBE](#) 

## Find Us on These Social Channels



# BRIDGING THE GAP

**Low-Power, Cost-Effective PIC32MM Family of Microcontrollers Offers Compelling 32-bit Solution for Applications with Budget, Power and Size Constraints**

## **First Family in Microchip's Portfolio of 32-bit MCUs to Offer Core Independent Peripherals**

Designers of today's embedded applications targeting the Internet of Things (IoT), consumer, industrial control, and motor control require flexible microcontrollers (MCUs) that consume less power, are more cost effective and have smaller form factors. To meet these challenging requirements, Microchip recently released a new family of low-power and low-cost 32-bit MCUs. The **PIC32MM** is our lowest-power and most cost-effective family of PIC32 MCUs, bridging the gap between our popular, low-power PIC24F and the low-cost PIC32MX families of devices.

These new, high-performance devices offer a wide range of features to suit your design's requirements. They operate from 2V–3.6V and are available in options with 16 KB, 32 KB and 64 KB Flash. For applications that require longer battery life in a smaller form factor, the PIC32MM devices offer sleep modes down to 500 nA and come in packages as small as 4 × 4 mm. They also incorporate Core Independent Peripherals—such

*(continued on page 5)*

as Configurable Logic Cells (CLCs) and Multiple-output Capture Compare PWMs (MCCPs)—that are designed to offload processing activities from the CPU for lower power and flexible system design. All of the digital peripherals are connected to Peripheral Pin Select (PPS) for easy pin mapping. This family also features a MIPS microAptiv™ UC core with microMIPS™ instructions, which combine 16-bit and 32-bit instructions for optimized code density. This enables these devices to achieve a 79 CoreMark™ score at 25 MHz operation. A shadow register set is included for faster interrupt response.

## Low Power

The PIC32MM family is optimized for low power consumption in battery-powered applications. It offers very-low-power sleep modes, which is where portable applications spend most of their time. The lowest-power sleep mode only draws 500 nA current. With a RTCC running, the sleep mode is less than 2  $\mu$ A.

## Low Cost

Despite the industry facing significant pricing pressure for general-purpose microcontrollers, the PIC32MM family offers a compelling solution for meeting your design's budget constraints. High-volume pricing starts as low as \$0.60 USD, and the published 10K unit pricing starts at \$0.76 USD.

## Small and Flexible Packages

Devices in the PIC32MM family are available in 20-, 28- and 36-pin packages, with form factors as small as 4 x 4 mm for 20-QFN or 28-uQFN options. And, while the 28-SPDIP for the 64 KB Flash version is not exactly small, it is easy for prototyping.

## Core Independent Peripherals

The PIC32MM is the first family in the PIC32 MCU portfolio to incorporate Microchip's unique Core Independent Peripherals. Designed to offload processing tasks from the CPU and reduce power consumption, these peripherals include a 12-bit Analog-to-Digital Converter (ADC), comparators, RTCC,

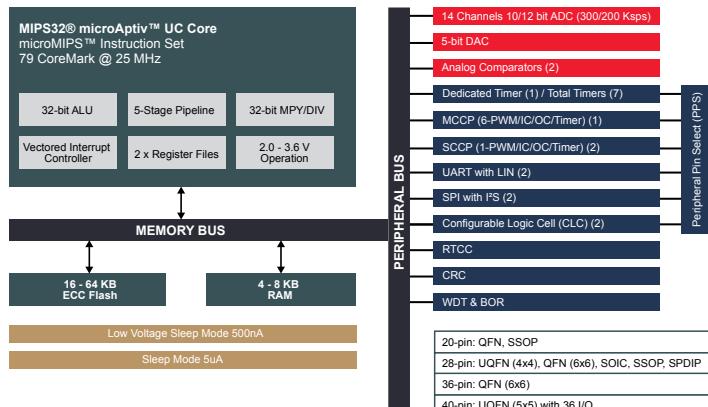


Figure 1 - PIC32MM Family Block Diagram

## COVER STORY



Watchdog Timer (WDT), Configurable Logic Cells (CLC), and flexible Pulse Width Modulation (PWM) and timer modules.

The Multiple Output Capture Compare PWM (MCCP) module can be configured in many different ways to provide up to six PWM outputs, or two 16-bit timers, or one 32-bit timer, or input capture, or output compare. The PWMs have resolution down to 21 ns and include output steering or complementary outputs. The MCCP can operate faster than the system clock and even operates in sleep mode. If you don't need the PWMs, you can use the MCCP as timers. This means that, when all the options are combined, this family offers a total of seven 16-bit timers

## Target Applications

With its low-power operation and small form factor packages, the PIC32MM family is a good fit for a wide variety of applications in the Internet of Things (IoT), consumer, industrial control and other markets. The flexible MCCP module helps enable low-cost BLCD motor control applications. Some other target applications include:

- **Low Power/Wireless Applications:** IoT sensor nodes, connected thermostats, environmental monitoring, remote controls, portable medical devices
- **Consumer Applications:** game consoles, home healthcare devices, fitness devices
- **Industrial Control Applications:** building automation, heating controls, lighting controls
- **Low-Cost Motor Control Applications:** white goods, table-top appliances

(continued on page 6)

# Supporting Ecosystem

A new hardware development tool,

## the PIC32MM0064GPL036 Plug-In

**Module** (MA320020), plugs into the

## Explorer 16 Development Board

(DM240001) to demonstrate the

capabilities of the low-power and

low-cost PIC32MM “GPL” family

of devices. The PIM includes the

PIC32MM0064GPL036—packaged in a 40-pin UQFN—which

has the most memory and the highest pin count of all devices

in this family. However, this PIM can be used for evaluation and

development for all members in the PIC32MM family.

The entire family of PIC32MM devices is also supported by the robust Microchip development ecosystem, including MPLAB® X Integrated Development Environment (IDE) and the MPLAB XC32 Compiler.

## MPLAB Code Configurator



When you're ready to get started with the PIC32MM family in your design, our **MPLAB Code Configurator**, a plug-in for the free MPLAB X IDE, enables easy peripheral setup, device



## COVER STORY

configuration and pin mapping. Visual screens make it simple to set up pin configurations and peripheral modes. The tool generates easy-to-read code with a single click, significantly reducing your software development time.

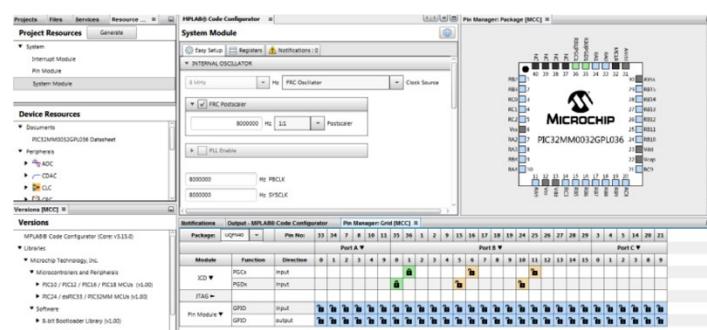


Figure 2 - MPLAB® Code Configurator Enables Easy Development with the PIC32MM Family of MCUs

Visit the **PIC32MM Family page** to learn more and for access to helpful resources that will help you get started with your design. Devices in the PIC32MM family are available now and can be ordered from **microchipDIRECT** or from **Microchip's worldwide distribution network**.



**Build something  
that matters**  
**HackadayPrize2016**



**Design Your  
Concept**



**Anything  
Goes**



**Citizen  
Scientist**

**For More Info Visit:**  
[hackaday.io/prize](http://hackaday.io/prize)



**Automation**



**Assistive  
Technologies**

Sponsored by:



# Taking on the World

Next-Generation Bluetooth® Low Energy Solutions  
Deliver Easy-to-Use Interface and Embedded  
Scripting Capability

## RN4870 and RN4871 Modules Support the Latest Bluetooth 4.2 Standard

Energy-efficient and economical Bluetooth Low Energy (BLE), also known as Bluetooth Smart, was created for the Internet of Things (IoT). This wireless technology is poised to power a wide range of connected applications, from wearable fitness devices to smart homes to the latest entertainment products to in-store advertising using beacons, and more.

If you are looking to build better end applications with lower power consumption, we are now offering two next-generation BLE solutions with an easy-to-use ASCII-style command interface that makes the devices easy to configure and eliminates any complicated code compiling. The **RN4870** and **RN4871** support the latest Bluetooth 4.2 specification and have a Bluetooth stack on board with a scripting engine to enable standalone operation and eliminate the need for a microcontroller (MCU) in simple applications. These next-generation

## NEW PRODUCT



Bluetooth devices enable a fast time to market and improve the total manufacturing cost for your design.

The RN4870 and RN4871 deliver up to 2.5 times the data throughput improvement over previous-generation products that are based on the Bluetooth 4.0 standard. In addition, these new Bluetooth solutions offer Federal Information Processing Standards (FIPS) based connection security coupled with advanced features. They can seamlessly transfer serial data over BLE devices and can support different beacon formats like iBeacon™ or Eddystone™ via a single command. Both devices, which also come with a shield option, are fully certified to meet worldwide regulatory standards and are available with compact package options as small as 6 x 8 mm. This, combined with a user-friendly interface, helps speed up your product development.

These modules are fully certified to meet worldwide regulatory standards.



The RN4870 and RN4871 deliver up to 2.5 times the data throughput improvement over previous-generation products.

These new RN-style modules are the easiest way to add our BLE solution to your products. Each module has an on-board software stack and pairs well with any of our large range of low-power microcontrollers or any microcontroller with a UART interface. The RN4870 and RN4871 are also excellent complements to our diverse low-energy solutions that target IoT applications.

(continued on page 8)

# Development Support

To help you get started with developing your BLE application, the **RN4870 Bluetooth 4.2 Low-Energy PICtail™/PICtail Plus Daughter Board** (RN-4870-SNSR), with a plug-in sensor board, works out of the box or can be interfaced to the **PIC18 Explorer Board** (DM183032), the **Explorer 16 Board** (DM240001 or DM240002), and the **PIC32 Expansion Board** (DM320002). Two iOS® applications are also available for use with the RN4870: Microchip's Smart Discover for iPhone® or iPad® and the Bluetooth Smart Sensor app.

The RN4870-V/RM118 is shielded with an on-board antenna and comes in a 12 × 22 mm module, while the RN4870U-V/RM118

## NEW PRODUCT

is unshielded with an external antenna and comes in a 12 × 15 mm module. The RN4871-V/RM118 is shielded with an on-board antenna and comes in a 9 × 11.5 mm module, and the RN4871U-V/RM118 is unshielded with an external antenna and offered in a 6 × 8 mm module. They can be ordered today from **microchipDIRECT** or from **Microchip's worldwide distribution network**. 

## MPLAB® Xpress Evaluation Board \$10 + Free Shipping

Good through 2/28/17



SAVE WITH  
COUPON CODE  
EXPINTRO16

**microchip**  
**DIRECT**

## PIC32MM Family for Low Power

Low Power and Low Cost in Small Packages

- Prices as low as \$0.60 in high volume
- Low sleep modes < 500 nA to optimize power consumption
- As small as 4 x 4 mm packages



# Now That Was Easy

Add Wi-Fi® Connectivity to Your Design with the ATWINC1500 Wi-Fi Network Controller

## No Network Experience Required to Accelerate Your IoT Product Development

**D**oes this sound familiar? You have a cool application, a slick design and a ready market. Except there's one final requirement: you need to add Wi-Fi connectivity and get that cool application connected to the Internet. Many developers get to this point and don't know where to begin.

Historically, adding Wi-Fi connectivity to an application was a daunting, tedious process best left only to RF engineers. Luckily, with the advent of the Internet of Things (IoT), the ubiquity and standardization of Wi-Fi, and the advancements in both hardware and software, it has become increasingly easy for developers to tackle Wi-Fi connectivity.

Continuing on with our goal of helping designers to easily add Wi-Fi connectivity to their applications, the **ATWINC1500** is our newest Wi-Fi solution. This IEEE 802.11 b/g/n IoT network controller SoC is an ideal add-on to existing microcontroller solutions, making it easy to bring Wi-Fi and network capabilities into embedded applications through UART or SPI-to-Wi-Fi interfaces.

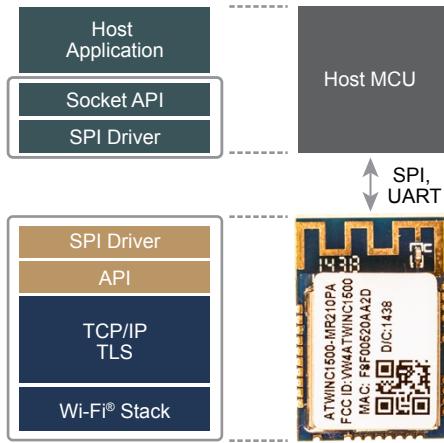


Figure 1 - ATWINC1500 Block Diagram

## NEW PRODUCT



As shown in Figure 1, the ATWINC1500 features a complete Wi-Fi and TCP/IP Stack, API and SPI driver on board and is designed to seamlessly connect via SPI or UART to the host MCU, which runs the application. The flexible ATWINC1500 connects to any microcontroller with minimal resource requirements.

The most advanced mode in the ATWINC1500 is a single-stream  $1 \times 1$  802.11n mode, providing up to 72 Mbps PHY of throughput. The ATWINC1500 features a fully integrated power amplifier, Low-Noise Amplifier (LNA), switch and power management modes. It also provides internal Flash memory as well as multiple peripheral interfaces, including UART, SPI, and I<sup>2</sup>C. The only external clock source needed for the ATWINC1500 is a high-speed crystal or oscillator with a wide variety of reference clock frequencies supported (between 12–32 MHz).

The innovative power architecture delivers ultra-low power consumption.

The ATWINC1500 uses an innovative power architecture that delivers ultra-low power consumption along with high performance. This approach reduces the number of external components and optimizes your bill of materials. In addition to ultra-low power, another key feature of the ATWINC1500 is its compact form factor. The ATWINC1500 comes in a tiny 5 × 5 mm QFN package for size-constrained applications.

(continued on page 10)

To help accelerate your design development even more, the ATWINC1500 is available as a single-chip module for fast integration ([ATWINC1500-MR210PB](#)). This highly integrated module has a small form factor (21.5 × 14.5 × 2.1 mm) while fully integrating a power amplifier, LNA, switch, power management and PCB antenna. The module is also fully certified, saving you considerable engineering time and certification costs.

The ATWINC1500 SoC and module are designed for a variety of target applications such as the IoT, smart appliances, multi-media streaming, safety and security, consumer electronics and industrial automation.

## Development Support

To get a quick start with developing applications with the ATWINC1500, the **XPRO Development Kit** (ATWINC1500-XPRO) is compatible with any existing Xplained PRO evaluation board. Additionally, the **ATWINC1500-XSTK Xplained Pro Starter Kit** (ATWINC1500-XSTK) can be used to demonstrate IoT setup, with LED control and temperature measurements sent wirelessly through the Internet to a mobile device.

## NEW PRODUCT

The ATWINC1500 40-pin QFN SoC, module, and development tools are available today from the [Atmel® website](#) or from [Microchip's worldwide distribution network](#). 



ATWINC1500-XSTK Xplained Pro Starter Kit (ATWINC1500-XSTK)

## Medical Embedded IoT/Cloud Solutions

Yeah, We've Got That



 **MICROCHIP**

# Music Minus Wires

Dual-Mode Bluetooth® Audio Devices Deliver Superior Sound Quality for High-End Headsets, Speakers and Sound Bars

## Enriching the Consumer Experience with the BM6x Family of Fully Certified Modules

With all the major advances in wireless audio technology that have developed over the past several years, consumers are not only expecting high-quality sound from their devices, but they are increasingly demanding a rich user experience that streamlines their communication and interaction with their audio equipment. Staying ahead of the competition in the wireless audio market means you need to find the right solution to combine the convenience of wireless, portable speakers with a simplistic interface to easily connect and control multiple end devices. As the next generation of dual-mode Bluetooth audio products from Microchip, the BM6x family of modules is designed to do exactly this.

The BM6x family builds on our successful BM2x portfolio of modules by introducing Bluetooth Low Energy (BLE) capability. Uniquely engineered for speakers, headsets and gaming headphones, this Flash-based platform offers ample flexibility and powerful design features, allowing you to easily

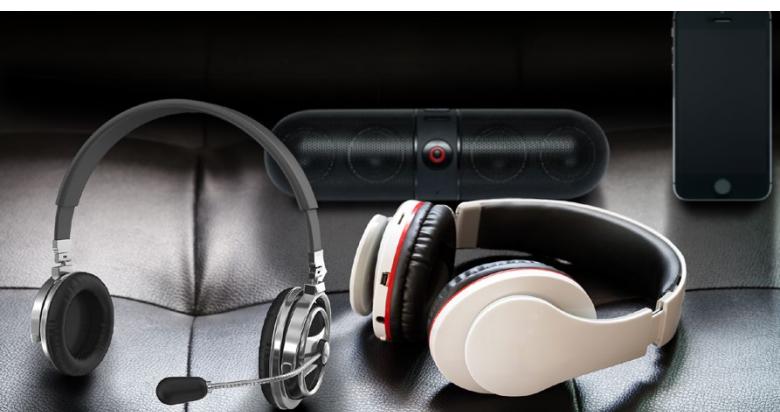


incorporate wireless connectivity in streaming music and voice command applications.

The combination of BLE and A2DP enables smartphone-to-speaker communication.

A 32-bit Digital Signal Processing (DSP) core provides the framework to develop sophisticated algorithms for advanced audio and voice processing. The 24-bit digital audio support delivers high-resolution audio to consumers for a richer listening experience. Sound systems comprised of multiple Bluetooth speakers benefit from ultra-low latency audio streaming, resulting in tightly synchronized audio playback amongst each speaker. Applications such as professional headsets benefit from high definition voice, achieved with a robust implementation of 16 kHz wideband voice with noise suppression and echo cancellation. The added firmware update capability allows for product software and configuration feature enhancements over time.

Qualified for Bluetooth v4.2, the BM6x family supports Enhanced Data Rate (EDR) links and the standard audio profiles. The powerful combination of BLE and Advanced Audio Distribution Profile (A2DP) enables smartphone-to-speaker communication



*The BM6x family of dual-mode Bluetooth® modules will help you stay ahead of the competition in the wireless audio market.*

**NEW PRODUCT**

*(continued on page 12)*

via a mobile app. Customized apps enrich the consumer's experience by providing creative control features such as pairing, remote control and real-time audio effect adjustments.

Our modules offer a turnkey solution for integrating wireless capability into your audio applications. You can choose from our **BM62SPKS1MC2-0001AA** or **BM64SPKS1MC2-0001AA** Class 2 devices or, if you are looking for extended range, choose our **BM64SPKS1MC1-0001AA** Class 1 device. All modules are fully certified with regulatory bodies in the United States (FCC) and Canada (IC), European Economic Area (CE), Korea (KCC), Taiwan (NCC) and Japan (MIC).

## NEW PRODUCT

## Development Support

Three evaluation boards are available to enable development of your wireless audio application using the BM6x family of devices: **BM-62-EVB**, **BM-64-EVB-C2** and **BM-64-EVB-C1**.

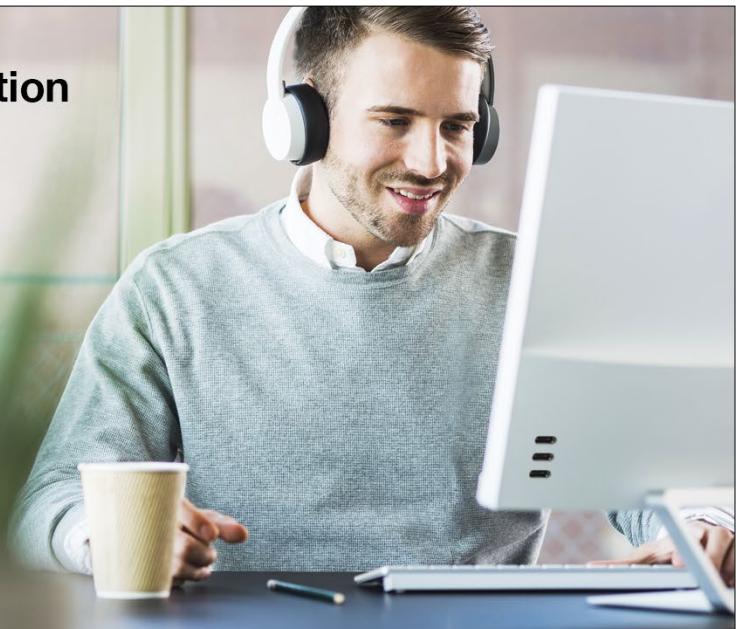
The modules and evaluation boards can be purchased from **microchipDIRECT** or from **Microchip's worldwide distribution network**. Visit our **Bluetooth page** to learn more about our products and technologies. 

Microchip's Custom Reel Service Now Available  
ORDER ANY QUANTITY UP TO A FULL REEL

**microchip**  
**DIRECT**

## Simple USB Type-C™ Implementation with USB 3.1 Smart Hub

Low-Cost Demo Board Gets You Started  
Implementing USB Type-C Right Away



 **MICROCHIP**

USB Type-C is a trademark of  
USB Implementers Forum.

# Get Ready to Innovate

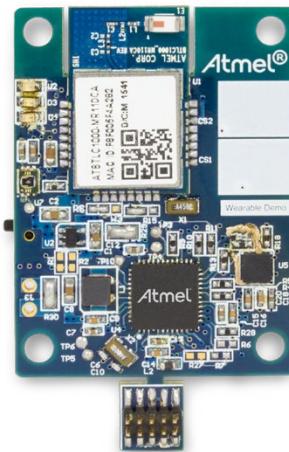
Delivering Over Four Times More Battery Life and Faster Time to Market, New Platform Includes MCU, Bluetooth® Low Energy and Sensor Technology

Based on Industry's Lowest-Power BLE Sensor Node for IoT Applications

Low power, smaller form factor and quick time to market are critical factors to making the billions of 'smart' and 'connected' things a reality in the rapidly growing Internet of Things (IoT) market. Next-generation IoT applications in personal health and fitness, indoor navigation, wearables, gaming, context awareness and augmented reality will require sensors to offer more immersive user experiences. Microchip is continuing to release new, innovative tools to enable more IoT developers to get their differentiated products to market faster.

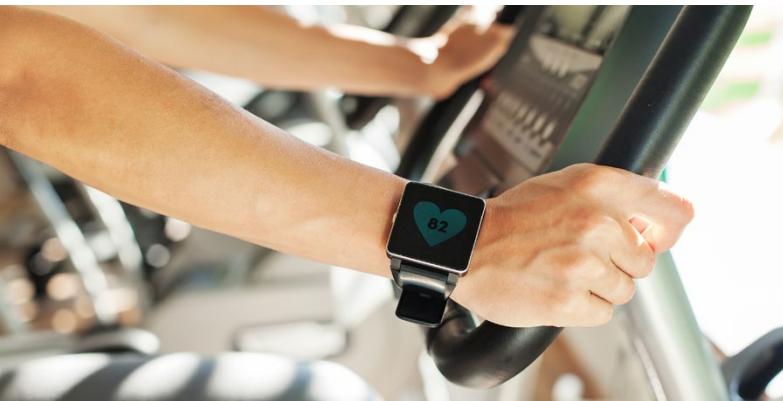
Our **Ultra-Low-Power Connected Demonstrator Platform** offers all the features to inspire the creation of innovative IoT applications. For designs requiring low power and smaller form factors, the kit integrates the **ATBTL1000-MR110CA** Bluetooth Low Energy (BLE) module that utilizes the world's lowest-power and smallest BLE solution. This module delivers at least 30 percent more power savings as compared to existing

NEW TOOL



solutions and is packaged in an ultra-tiny 2.2 × 2.1 mm Wafer Level ChipScale Package (WLCP). The platform also includes a **SMART SAM L21 Cortex®-M0+ microcontroller** (MCU) that achieves a ULPBench™ score of 185, the highest recorded score for any Cortex-M0+ while running the EEMBC® ULP-Bench, with power consumption down to 35 µA/MHz in active mode and 200 nA in sleep mode. By using these components, it is possible to increase battery life up to four times longer than solutions currently available on the market.

The platform includes a SMART SAM L21 Cortex®-M0+ microcontroller.



Create innovative, next-generation applications for the Internet of Things and get them to market faster.

The kit also includes the latest-generation Bosch 6-axis motion (BHI160) and environment (BME280) sensors that can be used for a wide variety of sensing applications. The BLE demonstration platform is an out-of-the-box solution that comes complete with source code, hardware design files, user guide and Android™ application source code that enable you to bring your product to market faster.

The Ultra-Low-Power Connected Demonstrator Platform (ATULPC-DEMO) can be purchased from the [Atmel® website](#) or from [Microchip's worldwide distribution network](#). 

# Weighing In

Add Intelligence and Bluetooth® Connectivity to Digital Weight Scales with Microchip's Flexible and Cost-Effective Solutions

NEW TOOL

## Reference Design Leverages 8-bit PIC® MCU and RN4020 Bluetooth Module

Digital weight scales have been used in homes, businesses and medical facilities for a number of years. As the demand for the ability to remotely monitor the health of patients continues to grow, connected weight scales are entering the rapidly expanding market for home-based consumer and medical devices. These connected scales can transmit data to smartphones, tablets and the cloud, where it can be used by caregivers and healthcare providers to help monitor the health and general wellbeing of their charges.

Reducing power consumption to extend battery life is a key concern for designers of digital medical weight scales. Other critical design considerations include high precision, fast response time and lower costs. Our PIC® microcontrollers with eXtreme Low Power (XLP) technology offer sleep currents as low as 9 nA and integrate features like op amps, ADCs, a USB controller, an LCD drive, touch channels and Core Independent Peripherals to offer extensive flexibility and performance for your weight

scale design. We offer a wide range of analog components to further support your project, and you can add wireless capability using our cost-effective and low-risk wireless controllers and modules. And, our free MPLAB® Code Configurator is integrated into MPLAB X IDE to offer a graphical programming environment that generates seamless, easy-to-understand C code for insertion into your project, speeding your development time and reducing your costs.

To help you get started, our Connected Weight Scale Demonstration Board shows you how to implement a low-cost, Bluetooth-connected weight scale using an eXtreme Low Power (XLP) 8-bit **PIC16F1783** microcontroller (MCU) and an **RN4020** Bluetooth module. This demo design can also be easily implemented using any of our Bluetooth Low Energy or Wi-Fi® connectivity modules.

The scale measures the user's weight in pounds and kilograms. Data and user instructions are transmitted via the Bluetooth module, demonstrating secure connectivity with mobile devices like tablets and smartphones. The design is powered by two AAA batteries and offers a low overall BOM cost due to function integration.

Visit the **Connected Weight Scale Demonstration Board** page on our website where you will find more details and resources. Start with the free download of the schematics and 'C' source code that can easily be modified to your specific application needs. Contact your local **Microchip sales representative** to see a working demonstration of Microchip's Connected Weight Scale demonstration board. 



Implement a low-cost, Bluetooth® connected weight scale with this reference design.



# Jump Start Your Internet of Things Designs

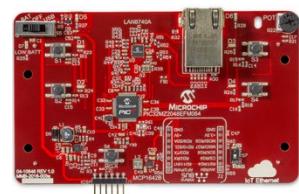
## Special Hardware Kits Combine Seamlessly with IoT Platforms from Microchip Cloud Design Partners

Connecting an embedded system to the cloud can be a daunting task. Not only do you need to have the right hardware solutions to enable your end product, you also need to define an efficient business model, navigate through the many options of cloud companies and then select the best one for your application and business. There are really two direct options, with the consulting piece serving as a third choice. The first option is to connect Microchip hardware to an Infrastructure as a Service (IaaS) like [Amazon Web Services](#). The second option is to use an IoT platform provided by Software as a Service (SaaS) companies. To address the IaaS-based option and show both IoT startups and established companies how to connect a 32-bit microcontroller to the AWS IoT, Microchip has developed an Ethernet solution based on a PIC32 MCU. In addition, Microchip has partnered with some leading Internet of Things (IoT) Software-as-a-Service (SaaS) solutions providers to reduce your time to market, simplify your development and help you get your IoT design quickly connected and running on the cloud.

### IoT Ethernet Kit Powered by AWS IoT

We are proud to be part of the partner network for the Amazon Web Services (AWS) managed cloud platform. This scalable and global platform enables you to easily and quickly connect devices to the AWS IoT service, where they can interact with

cloud applications and other devices securely. The [IoT Ethernet Kit](#) (DM990004) comes preloaded with the AWS IoT firmware to get your example application up and running within minutes. It shows you how to connect a 32-bit PIC® microcontroller to an IaaS using our low-power LAN8740A 10/100 Fast Ethernet Physical Layer Transceiver (PHY) driven by a 32-bit PIC32MZ EF microcontroller (MCU) with 2 MB of Flash. An on-board mikroBUS™ connector allows you to add any of the sensors available on MikroElektronika's vast selection of click boards™. The kit operates with FreeRTOS™ running the MQTT lightweight messaging protocol and TLS security stack from WolfSSL, which provides you with the necessary resources to start designing your IoT project.



**IoT Ethernet Kit**  
(DM990004)

### IoT Ethernet Monitoring Kit with Medium One and Saritasa

Get a solid starting point for your application with the [IoT Ethernet Monitoring Kit](#) (DM990101). Microchip has partnered with [Medium One](#) and [Saritasa](#) to provide this end-to-end

(continued on page 16)

reference solution that enables you to prototype, test and connect products to the cloud and to efficiently access, analyze and visualize data. The kit includes the IoT Ethernet Monitoring Board with a LAN8740A PHY, a PIC32MZ EF MCU and support for four sensors using

MikroElektronika click boards: temperature/humidity, air quality, pressure and motion. Cloud services from Medium One—an IoT cloud data intelligence company—are embedded on the board to allow you to rapidly build your application. A production-ready iOS® mobile app from Saritasa—an IoT system integrator, providing firmware, software and smart mobile app development—also comes with the kit. An open reference design is also included. You will find more extensive details about this kit in the “Unifying the Internet of Things” article on [page 17](#).

## WCM Wi-Fi® Client Module Development Kit Supported by ExositeReady™



WCM Wi-Fi® Client Module Development Kit (DM182023)

Fast track your IoT initiatives with the [WCM Wi-Fi Client Module Development Kit](#) (DM182023) and [Exosite](#), a leading provider in the IoT platform market. The kit's development board incorporates an MRF24WG0MA Wi-Fi Module and a PIC32MX microcontroller and is the first hardware platform certified by the [ExositeReady Embedded Software Development Kit](#)

(SDK) to give embedded software developers access to a comprehensive set of open-source tools for use with Exosite's advanced cloud-based IoT platform. The kit comes preprogrammed from Microchip's factories with the ExositeReady cloud agent to enhance the out-of-the-box design experience and to enable faster prototyping and evaluation for a variety of applications. The ExositeReady program includes a complete suite of software modules, ports, instructions and examples. This ensures that you can efficiently develop cloud-connected,



MEDIUM ONE  
IoT Ethernet Monitoring Kit  
(DM990101)

## DESIGN CORNER

production-ready embedded devices with less risk and investment. Visit the [Microchip ExositeReady Platforms web page](#) to find a number of resources to help you get started with your Wi-Fi-based IoT project.

## PIC32 Ethernet Starter Kit and PubNub

[PubNub](#) is a global data stream network that simplifies the deployment of IoT projects by enabling the remote control and monitoring of devices. The [PIC32 Ethernet Starter Kit](#) (DM320004) provides

the easiest and lowest-cost method to experience 10/100 Ethernet development using PIC32 microcontrollers. This board allows you to easily add real-time, bidirectional communication to your PIC32 MCU-based project by using the [PubNub](#)



**PubNub®**

PIC32 Ethernet Starter Kit (DM320004)

[PIC32 client library](#) for the MPLAB® Harmony integrated software framework. Whether you are using a PIC32 MCU in a home automation, industrial IoT or smart city application, PubNub will let you easily scale that application to millions of users and 75+ platforms (mobile, web and IoT) with minimal battery drain and bandwidth consumption. You can implement critical features including remote device control, secure firmware upgrades and plug-and-play device provisioning. PubNub's advanced features include device status and metadata monitoring, fine-grain access control, message storage and playback, and automatic data catch-up on unreliable networks. Read the [Getting Started with the Microchip PIC32 Microcontroller tutorial](#) on PubNub's blog to learn more.

As an IoT powerhouse, Microchip is committed to providing the resources and support you need to get your embedded design quickly, easily and safely connected to the cloud. Visit our [Internet of Things Design Center](#) to discover how our portfolio of solutions can help you get started. 

# Unifying the Internet of Things

Microchip Partners with Medium One to Provide a Complete IoT Solution that Easily Integrates Connectivity and Sensors with Any System

Connecting an existing “thing” to the cloud sounds great, but where do you start? Maybe you are making a vending machine and would like to buy sensors to monitor its electrical current for energy usage. Or maybe you already have a piece of industrial equipment with sensors, but you need a way to collect the data and get it off the board and onto your servers for operations management to access. There are so many things to consider: security, identity management, dashboards, analytics, third party access and much more.

Medium One, an official Microchip Cloud Design Partner, is an Internet of Things (IoT) cloud data intelligence company. The company’s cloud services and real-time workflows allow you to rapidly build IoT applications. All the functions needed for development are included, from connecting to the cloud and processing device data to IoT-centric libraries with machine learning and analytics functions for predicting trends and detecting

anomalies. Medium One aggregates all event data from any source—whether it is from sensors, mobile apps or the existing cloud—and performs real-time processing to extract intelligence or implement automation. All data is handled in a secure environment with robust multi-layered security features. This allows you to focus on your application logic while Medium One quickly extracts intelligent insights to power your design.



As noted in the “Jump Start Your Internet of Things Design” article on [page 15](#), Medium One has partnered with Microchip and Saritasa to offer the **IoT Ethernet Monitoring Kit**. This kit brings together a powerful PIC32MZ EF microcontroller, Ethernet-based connectivity (LAN8740A) and a wide range of technologies seamlessly to provide a secure and scalable

*(continued on page 18)*

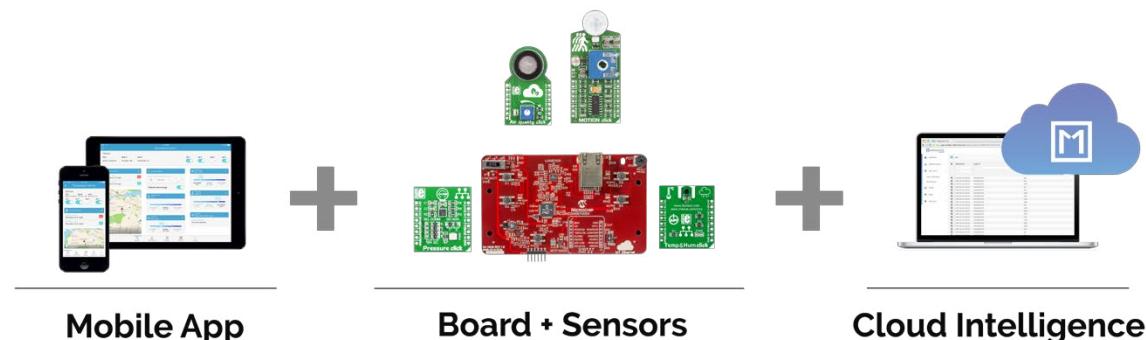


Figure 1 - Components of the IoT Ethernet Monitoring Kit

embedded, cloud and mobile framework to implement any IoT design quickly with no monthly fees or programming required.

## DESIGN CORNER

## Embedded

Offered as an out-of-the-box solution that allows you to connect to the cloud in minutes, the IoT Ethernet Monitoring Kit is preflashed to prevent the need for any embedded programming. However, an **MPLAB® Harmony project** is available to assist with custom development if needed. The board comes with a MikroBus™ socket that supports prebuilt drivers to work with multiple click boards™ from MikroElektronika including the Air Quality Click, Motion Click, Pressure Click and Humidity/Temperature Click, which are sold separately. A reference design is also included to provide an excellent starting point for modifying or adding your choice of click boards to your product if you already have access to sensor or device data.

Single-click IoT provisioning is available once the kit is activated. After the board is plugged into the same network as your browser, the board page can be accessed by a unique URL provided by Medium One during setup. The provision service accepts the board's credentials and name and also specifies which sensor is connected. This web-based provisioning is customizable to support production applications.

Once the board's identity is established on the cloud, it can use the MQTT protocol with TLS security for all communications. It can publish information from sensors, buttons and potentiometer data, as well as publish data on a regular basis to determine if the board loses its connection. Device memory and other device statistics that perform remote diagnostics can also be tracked. MQTT subscriptions are used to retrieve and synchronize the state of the board with the cloud. The board also dynamically loads its unique sensor configuration from the cloud upon boot up; this includes specifying the sampling frequency and interrupting the threshold for sensors to optimize the network traffic for a specific device or application. Messages can easily be received from the cloud to control another peripheral. For

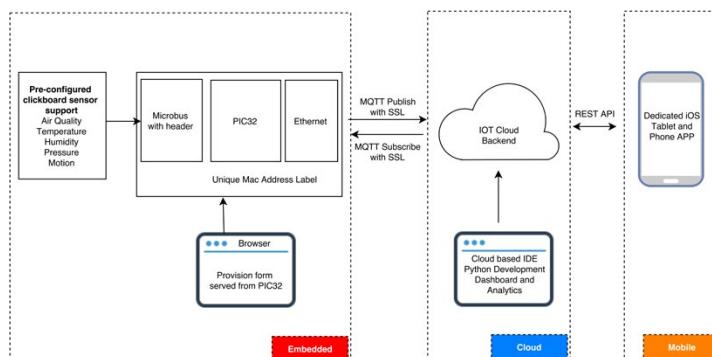


Figure 2 - IoT Ethernet Monitoring Kit Block Diagram

example, the board's LED lights can be controlled via the cloud and the kit's mobile app.

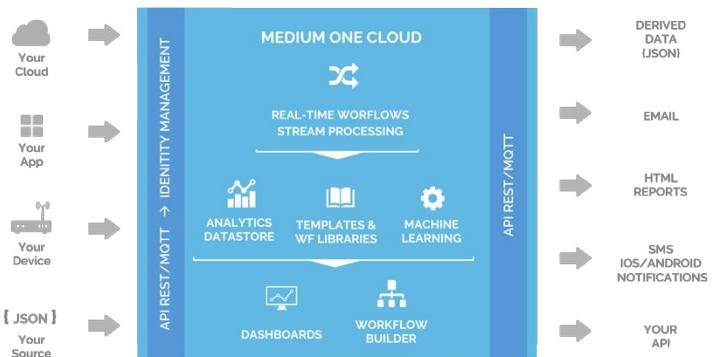


Figure 3 - Medium One Cloud Services

## Cloud

Medium One's cloud stack offers services ranging from identity management to storage, providing the perfect environment for building real-time applications. The cloud platform can be managed by Medium One or deployed in a custom infrastructure. A variety of API integration options is available to support scalability in any environment.

A cloud-based Python IDE allows you to implement real-time cloud applications using the built-in data analytics and IoT libraries. The Python code is open so that it can be modified and customized. The cloud logic is natively designed to process 1:1 for each device, allowing you to focus on your application instead of on device segmentation. It also supports grouped analytics to gather aggregated data that can be used to create reports or make real-time decisions.

Medium One offers the following pre-built workflows designed specifically for the IoT Ethernet Monitoring Kit:

- Real-time sensor monitoring
- iOS® push notifications
- IP-based location tracking
- Custom rules engines
- Device and identity management
- Offline monitoring
- Anomaly detection
- Daily and weekly email reports
- Device state synchronization

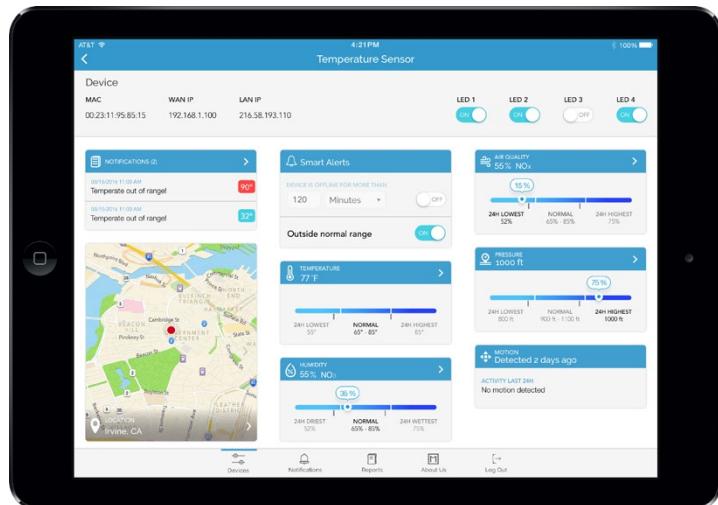
(continued on page 19)

## Special Features

**Geo-locationing:** Check if a region has changed by tracking the location of a device. An alert is generated if the IP address or location has changed above a given threshold. It can also detect if the device is offline and provide daily statistics on the percentage of time the device is connected to the Internet.

**Sensor Intelligence:** A variety of data and sensor intelligence is available, including alerts if sensor readings are outside normal sensor ranges. The platform learns the normal range based on the day of the week and time of day that the sensor picks up the data.

**Roles and Permissions:** The necessary roles and permissions have already been established to make this kit nearly production ready. These include separate identities and roles for the device, mobile users, cloud administrators and developers.



Monitor connectivity and sensor data with the iOS® mobile app.

## DESIGN CORNER

### Mobile

The production-ready iOS mobile app was designed by Saritasa and enables you to monitor the connectivity and sensor data in real time. It integrates with the Medium One cloud services via REST and displays all devices connected to your project. It also controls the sensor LEDs by sending commands to the Medium One cloud, which relays the data to the board via MQTT. Pre-built visualization widgets are available to display history and trends, as well as to create rules and alerts. These widgets can easily be extended to be used with other sensors and data types. The mobile app is designed to be white-labeled and customizable for most applications.

### What Can You Do with Your Custom Cloud Application?

- Create your own analytics and metrics for your business applications
- Customize your own reports and alerts
- Monitor your own data stream and sensors
- Gather insight across your connected devices
- Integrate with your existing back-end infrastructure through the open API
- Request machine learning models to get prediction or recommendation for your application

If you are ready to get started with your smart monitoring project, visit the [Introducing the IoT Ethernet Monitoring Kit](#) page on Medium One's website to learn more. 



**SUMMER DEV TOOLS DEALS!**

GET UP TO 50% OFF

CLICK FOR DEALS

# Solving the Security Conundrum

## How Hardware Cryptography Solutions for Securing Internet of Things Nodes Can Minimize Your Embedded Design Costs

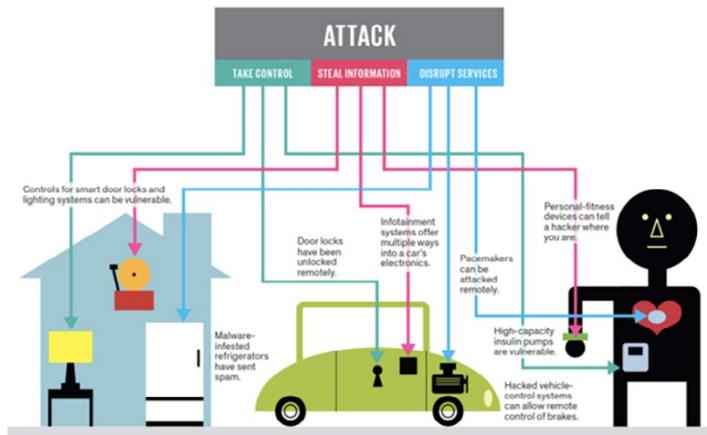
Securing the Internet of Things (IoT) is a hot topic right now, especially as new instances of security breaches come to light. Stories of hackers taking control of a vehicle that was cruising on the highway at 70 miles per hour, of a flying plane being hacked via its in-flight entertainment system, and of smart meters in Spain being compromised for energy theft have helped to make IoT security a huge concern for consumers as well as embedded designers. The value and benefit of smart, connected devices has been overshadowed by uncertainty about the security issues that may come with them.

Security requirements for the IoT are radically different from those of traditional embedded designs because small, intelligent IoT nodes must be able to authenticate themselves to

the ecosystem with limited or no human interaction. In order to do this, their cryptographic credentials must be stored in the device itself. Since edge nodes are often physically accessible, these credentials are in danger of exposure to hackers.

There are three major pillars for securing IoT designs: authentication, encryption, and secure key storage. While authentication and encryption are equally important building blocks in the security equation, it's important to recognize that encryption doesn't ensure security in IoT applications. Since it is generally too daunting and time consuming to hack into complex authentication and encryption systems, the majority of hacks are usually implemented by using imposter devices to break into the areas where secret keys are stored. The loss of these secret keys severely impacts the authentication and encryption processes. In other words, once hackers get a hold of the keys, the other two protection mechanisms can also be compromised or breached. Therefore, an IoT system is really only as strong as the process that is used for the storage of secret keys.

To address this challenge, embedded design engineers have two options for implementing security in their IoT projects: software-based and hardware-based solutions. While software-based solutions have the advantage of flexibility, they generally don't meet the needs of IoT systems. Hackers and spoofers can exploit software bugs to break into a network of devices. Updating the software with patches is a stopgap



*The diversity of the IoT world demands a new security paradigm.*

*(continued on page 21)*

measure that sometimes only offers a temporary solution. On the other hand, hardware-based systems are far more difficult to spoof or hack than software-based systems, making them a more complete security solution.

## The IoT's Hardware Security Blueprint

In IoT designs, the authentication and cryptography security elements need to be implemented fairly early in the development cycle. Therefore, it is critical that IoT developers have a clear understanding of an end-to-end ecosystem for creating a robust security roadmap. The traditional approach to hardware security involves building the security stack on top of a microprocessor (MPU) or microcontroller (MCU). However, a significant amount of computing power is required to carry out security operations. For instance, precious CPU cycles are required to accelerate the authentication of applications and firmware. As a result, hardware solutions built around a central MPU or MCU have seen limited success because this compute-intensive authentication burdens the overall system and eventually slows down the device performance. Also, security solutions built on top of processors generally store the secret keys into memory, which can be accessed by clever hackers via intrusion mechanisms like snooping. Security co-processors or crypto elements can help resolve this design conundrum.

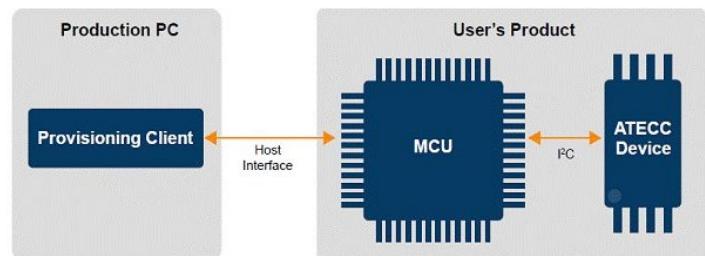
## Enter Security Co-processors

A security co-processor is a separate chip that offloads security tasks from the main microprocessor or microcontroller while making authentication faster and less taxing on the overall product design. Its crypto elements offer protection against a wide array of security threats including IP theft, cloning, over-building, reverse engineering and infecting IoT devices with rogue code.

An IoT node or gateway is the central nervous system that handles a horde of sensors and other “things” at one end and

## DESIGN CORNER

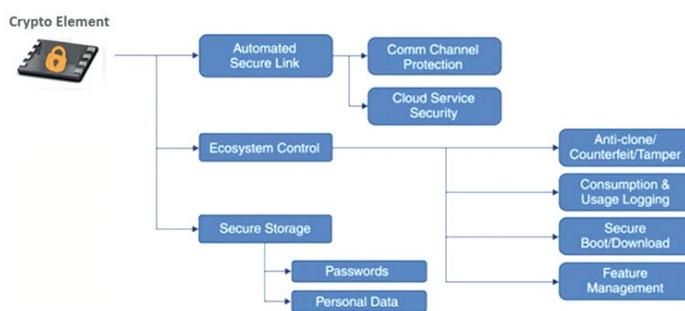
resource-sensitive cloud computing at the other end. It can only be truly secure when the secret keys are stored in protected hardware. Using security co-processors in a node or gateway design will safeguard private keys, certificates and other sensitive security data to ensure authentication and protection against software, hardware and back-door attacks.



*A security co-processor or crypto element offloads the authentication tasks from the main MCU*

A security co-processor is a cost-effective addition and its crypto elements consume very low power, which is essential in IoT designs. Some other reasons to consider using security co-processors include:

- They are equipped with hardware cryptographic acceleration to carry out strong authentication while they offload cryptographic code and math from the central processor of an IoT node
- They simplify the manufacturing process by effectively dealing with the key provisioning, a process of loading security keys and other security features like certificates onto a product
- They are pre-loaded with unique keys and certificates to ensure that encryption keys are secure throughout the manufacturing process, which also reduces complexities in manufacturer supply chains
- They replace the traditional approach of using the Hardware Security Module (HSM), which uses a database approach to store, protect and manage keys, as well as requires a significant investment in infrastructure and logistics
- They can also replace Trusted Platform Module (TPM) chips, which have their roots in the computer industry and are not well-positioned for low-price IoT edge applications; instead security co-processors facilitate certified ID platforms and can be deployed in IoT node-to-cloud environments, even at smaller scales

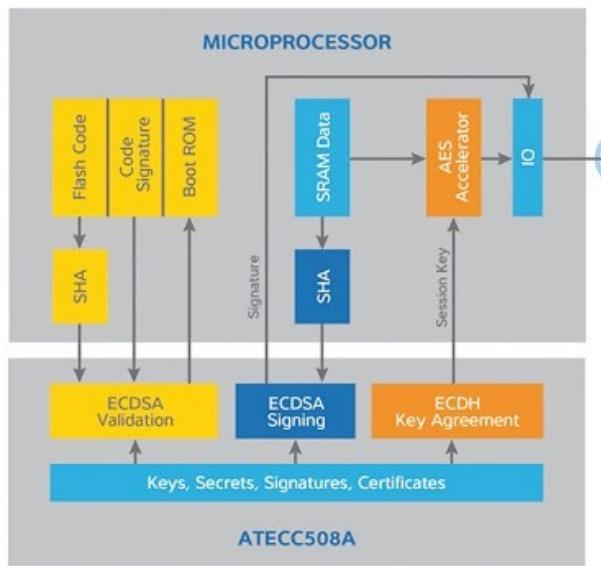


*A crypto element allows small, low-cost IoT nodes to implement strong cryptographic security*

*(continued on page 22)*

# A Complete IoT Security Platform

Since the IoT security paradigm is somewhat of a moving target—with multiple device types communicating with each other over a multitude of wired and wireless interfaces—IoT developers are relying on the expertise of chip suppliers that have years of experience in crypto hardware and security standards like TPM and Secure Hash Algorithm (SHA). Microchip not only offers a range of security co-processors that are compatible with any MPU or MCU, but also powerful development tools to help you implement security co-processors more quickly into your embedded systems.



The ATECC508A crypto element first generates its own key and then carries out ECC operation to authenticate IoT nodes

## DESIGN CORNER

Primarily targeting cost-sensitive embedded applications, the **ATSHA204A** crypto element is based on SHA-based symmetric authentication and uses 4.5 Kbytes EEPROM. The **ATAES132A** crypto element, based on the Advanced Encryption Standard (AES), boasts 32 Kbytes of secure serial EEPROM to facilitate faster symmetric authentication.

The **ATECC508A** crypto engine uses Elliptic Curve Cryptography (ECC) to create secure, hardware-based key storage for IoT applications that include home automation, industrial networking and medical. It employs protected key storage as well as hardware acceleration of ECC cipher suites, including mutual authentication with Elliptic Curve Digital Signature Algorithm (ECDSA) and key agreement with Elliptic Curve Diffie-Hellman (ECDH) techniques. It also ensures secure storage by utilizing the **Hardware-TLS** (HW-TLS) platform. The ATECC508A supports the **CryptoAuth Xplained Pro** evaluation and development kit and works with the **CryptoAuthLib** software support library. Proving that hardware-based security solutions do not need to be costly, the ATECC508A minimizes IT labor costs and licensing fees and is offered at very attractive prices for high-volume IoT applications.

If you need world-class and cost-effective embedded security solutions for your IoT system design, visit the **Atmel® Security ICs** web page to learn more about how our hardware-based security devices can reduce your costs and ensure trust for every system design. 



**microchip**  
**DIRECT**



## Rapid Prototyping, Seamless Integration

Ready to meet the many needs of today's embedded connectivity projects, PIC32 microcontrollers (MCUs) offer a powerful combination of best-in-class performance, larger memory configuration and connectivity peripherals. **MPLAB Harmony** is a flexible, abstracted, fully integrated firmware development platform that offers system services, device drivers and other libraries and is designed specifically for use with PIC32 MCUs.

MPLAB Harmony's software framework includes a number of **connectivity building blocks** that will speed your development of embedded connectivity applications. These easy-to-use and highly configurable Ethernet, Wi-Fi®, USB, Bluetooth®, CAN and Serial modules work together seamlessly, allowing you to add or remove functionality with little or no change to your application code. These connectivity blocks abstract away device-specific details so that they can be ported across multiple PIC32 devices.

MPLAB Harmony also features the MPLAB Harmony Configurator (MHC) plug-in, a powerful configuration tool that enables you to add these connectivity software building blocks with just a point and click of the mouse. MHC also provides numerous configuration options for each software building block to meet your specific design needs. These options can be easily selected with a click of a mouse, whereas if you were to use a traditional embedded development environment, you would need to write code and test for every configuration option your design might require. These MPLAB Harmony features provide PIC32 MCU users with the benefit of rapid prototyping

for various connectivity applications including USB and TCP/IP, which require interpreting complex protocols and significant processing. This will drastically reduce your development time and effort, resulting in a faster time to market.

To help you get started with developing your PIC32 MCU-based embedded connectivity design using MPLAB Harmony, from now through December 31, 2016, we are offering a 25% discount on the **PIC32MZE Starter Kit** (DM320007) when you order these development tools from microchipDIRECT using coupon code MX57MZE.



**PIC32MZE Starter Kit**  
(DM320007)

Also, visit our **32-bit Microcontroller page** where you can discover the device with the right processing power, memory and peripherals for your design and find additional resources to help you get started with your next embedded design.



## Custom Control

### Implement a High-Resolution Pulse Width Module by Combing Two Core Independent Peripherals

The Core Independent Peripherals available in a number of PIC® microcontrollers provide many opportunities for you to simplify your designs while also creating more innovative applications and products. By taking the processing load off the CPU, Core Independent Peripherals allow smaller, lower-power PIC MCUs to perform extremely complex tasks. For example, although many applications can function with lower Pulse Width Modulation (PWM) resolutions, higher resolution is needed for the dimming of LED lamps because of the human eye's sensitivity to flicker.

Conventional PWMs start losing effective resolution at relatively low switching frequencies. The smallest incremental change in pulse width achievable by a conventional PWM with a 16 MHz system clock speed is 62.5 ns (one system clock). If the fastest available PWM clock is FOSC/4, then this increases to 250 ns.

If your application requires fairly high switching frequencies that deliver as much PWM resolution as possible at these frequencies, you can combine two Core Independent Peripherals—the Numerically Controlled Oscillator (NCO) and the Configurable Logic Cell (CLC)—to implement a high-resolution PWM output and add more functionality and flexibility to your design. Using this technique, you can create a PWM with an incremental average pulse width change of as little as 15 ps.

Even if high resolution is not the primary requirement in your design, this solution may still be attractive for a number of applications. It allows you to incorporate an additional PWM into your design or create a constant on/off-time variable-frequency PWM, where the pulse is triggered externally as required. An example of this would be performing zero current switching in high-efficiency power converters.

The NCO and CLC are available in the following 8-bit PIC MCU product families:

Product Family	Pin Count
PIC10(L)F3XX	6
PIC16(L)F150X	8–20
PIC16(L)F171X	28–40
PIC16(L)F183XX	8–20
PIC16(L)188XX	28–40

To learn more about how to implement a high-resolution PWM using the CLC and NCO, download [Application Note AN1476: Combining the CLC and NCO to Implement a High-Resolution PWM](#). Visit our [Core Independent Peripherals page](#) to discover other ways to take advantage of these flexible and powerful on-board modules. 

# What Is the True Cost?

## Making the Wisest Choice Between Thermistors and Integrated Temperature Sensors

Temperature is still the most commonly measured physical phenomenon, with users spending an estimated five billion dollars on temperature monitoring solutions in 2016. Of those dollars spent, nearly twenty percent went to thermistor-based solutions. This article will take a quick look at why thermistors are so prolific for temperature sensing applications, as well as discuss the benefits of using other silicon-based temperature monitoring solutions.

A thermistor is a ceramic semiconductor material typically coated in glass. This device acts as a temperature-dependent resistor, meaning the resistance changes as a

function of temperature. This is similar in concept to a Resistive Temperature Detector, or RTD. However, unlike an RTD, which is simply formed from a pure metal wire, a thermistor is a semiconductor material and, as such, is limited in terms of temperature range and subject to process variations.

So why are these thermistors so prolific? Relative to other, comparable solutions such as silicon-based integrated circuits, thermistors are available in a wide variety of packaging options, including very small packages. This small packaging also has the benefit of fast thermal response times. But the main reason people use thermistors is money. Thermistors are readily available and appear to be much cheaper than a comparable voltage output integrated circuit temperature sensor. But what is this cost-effective solution actually costing you?

As mentioned previously, a thermistor acts as a temperature-dependent resistor, but this resistance is highly non-linear when measured over a wide temperature range. Figure 1 shows the response of a typical thermistor across temperature. From 0°C to around 50°C, the change in resistance as a function of temperature is fairly linear, with higher temperatures resulting in extremely small changes in resistance that can be very difficult to measure accurately. To make matters worse, this non-linear behavior is dependent on variations in the semiconductor process, which must be characterized by that specific manufacturer.

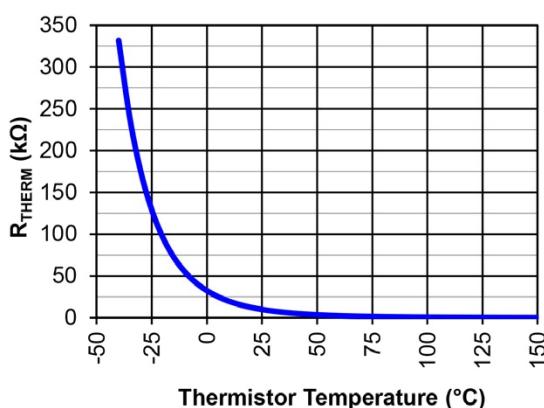


Figure 1 - Non-linear Resistance of Typical Thermistor

(continued on page 26)

On the other hand, a voltage output integrated circuit temperature sensor, such as the **MCP9700A** or the **MCP9701A**, will provide a linearized output across the entire operating temperature range, such as 10 mV/°C or 20 mV/°C, making it much easier to accurately measure with an Analog-to-Digital Converter (ADC).

In order to sense this change in resistance across a thermistor, it must be biased, typically by creating a simple voltage divider, as shown in Figure 2. The supply voltage, shown as +V, will typically be used as the reference voltage for the associated ADC, creating a ratiometric measurement that removes any error associated with variations in the supply voltage. The value of the top resistor, noted here as  $R_A$ , must be carefully considered for the given application.

In the above example, a nominal 10 kΩ thermistor is used. By adjusting the value of the resistor  $R_A$ , one can adjust the most linear range of the thermistor, moving it up or down the temperature scale, as shown in Figure 3.

When the resistance of  $R_A$  is reduced, the linear region moves up in temperature. However, as the value of  $R_A$  goes down, the current through the resistor divider network increases. This has two negative consequences. The obvious one is that more energy is expended, leading to reduced energy efficiency. The less obvious consequence is self-heating of the thermistor. As the current through the thermistor increases, it can

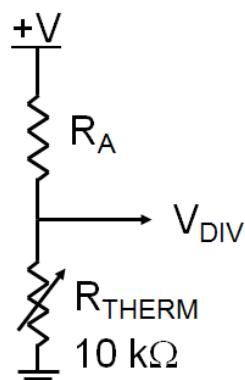


Figure 2 - Voltage Divider

## DESIGN CORNER

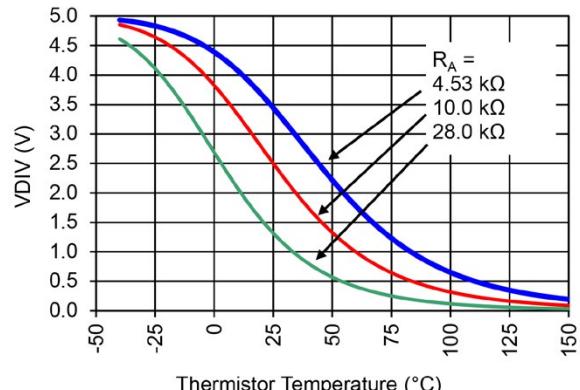


Figure 3 - Adjusting Linear Range of Thermistor

cause the temperature of the device to rise above that of the ambient temperature, adversely affecting the accuracy of the solution. A voltage output integrated circuit does not have these issues, as the supply current needed to power these types of devices is typically very small, on the order of tens of micro amperes or less, which makes self-heating a non-issue.

So the next time an application calls for temperature sensing, be sure to consider the trade-offs associated with thermistors. They can provide a cost-effective solution, but what is the true cost in terms of accuracy, temperature range, complexity and power usage? Depending on the application, a voltage output temperature sensor may not be so expensive after all. For more information about our thermal management products, visit the [Temperature Sensor/Fan Control area](#) of our website. 

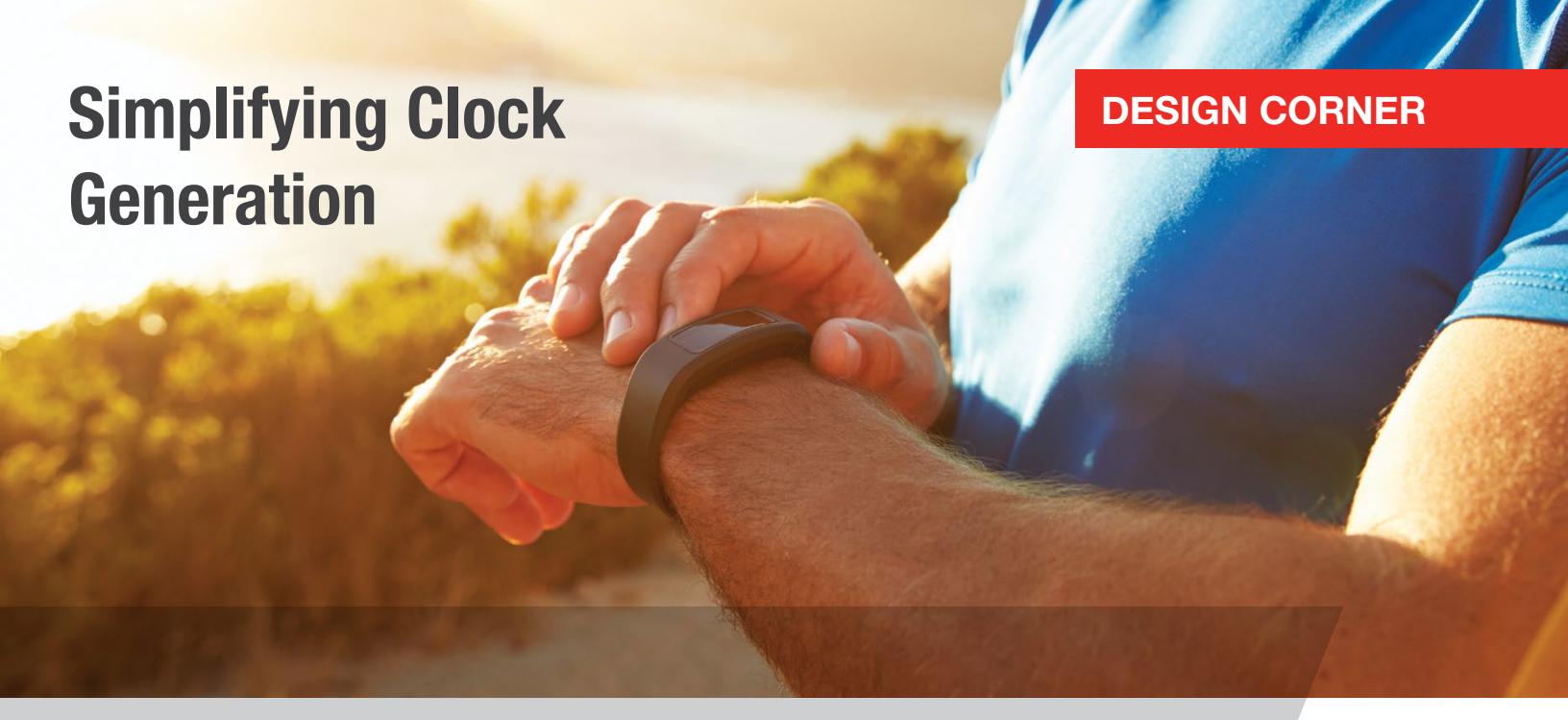
**Thrive at Extremes**  
Robust and Reliable Ethernet

**MICROCHIP** KSZ8061 **MICROCHIP** LAN9252

**MICROCHIP**

# Simplifying Clock Generation

DESIGN CORNER



## Reduce Cost and PCB Footprint with Phase-Locked Loop-Based Frequency Translator ICs

Phase-Locked Loop (PLL) based frequency translator ICs, like our **PL611s-02**, **PL613-21** and others, can be driven from a single frequency-reference source to generate the multiple frequencies required in a complex clock tree. This eliminates the need for multiple independent frequency references, saving cost and board space. Most often these devices are offered as programmable integer-N PLL ICs.

The functional block diagram of an integer-N PLL is shown in Figure 1. The input frequency to the PLL IC ( $f_{IN}$  in the diagram) can be from a crystal, MEMS resonator, or from another frequency source that is already on the board.

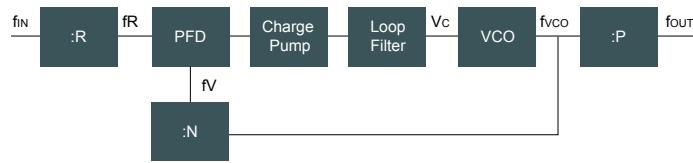


Figure 1 - Basic Integer-N PLL Functional Diagram

A programmable integer-N PLL IC can include one or several of these PLLs. In principle, all elements of a PLL can be programmable. In practice, the programmability is usually limited to the R, N, and P counters, and sometimes the charge pump current and the loop filter parameters. In addition to elements of the PLL, some other functions of the IC can be programmable, like the strength of the output drivers (e.g. the **PL671-01**), or the ability to switch between various programming configurations

to meet different industry standards in one clock IC (e.g. the **PL613-01** and **SM802XX**). This article primarily focuses on the design trade-offs related to R, N, and P counter selection for optimum performance.

The programmability is usually provided by on-chip One-Time Programmable (OTP) memory. Programming software like the **Microchip ClockWorks® Configurator** is available to assist you in configuring/programming the IC at the sample stage. Using this tool, you only need to provide the desired input and target output frequencies, power supply information (1.8V, 2.5V, 3.3V), the output driver strength, pin configuration, etc., to program the clock IC. Some devices, like the SM802XX and **SM803XX** series, allow some of the IC to be reconfigured on the fly (during operation) using the SPI interface, for example.

Usually the output frequency is set by the target application. There is some freedom for setting the input frequency, but in many cases it is also set by the application. The PLL converts the input frequency ( $f_{IN}$ ) to the output frequency ( $f_{OUT}$ ). The integer PLL does the conversion with some limitations. You will need to take these limitations into account when you specify the tolerance of the output frequency of the PLL. The equation for the output frequency of the integer-N PLL demonstrates the nature of these limitations:

$$f_{OUT} = \frac{N}{P \cdot R} f_{IN}$$

(continued on page 28)

The output frequency can only take discrete values. If the ratio between  $f_{IN}$  and  $f_{OUT}$  is an irrational number, the integer-N PLL will have to approximate this number by using a rational number. At first glance, it appears that this is not a problem because you can get close to  $f_{OUT}$  by increasing N, P, and R. However, it turns out that there are limitations on the values of dividers in the integer-N PLL. Some of these limitations are apparent. Counters have only a limited number of bits and the Voltage-Controlled Oscillator (VCO) frequency, which is directly related to the values of counters, has low and high limits. Some of the limitations are not apparent, but they are related to the PLL being a feedback system that can be unstable with some combination of divider values. Here are some factors to consider:

- The N counter value is limited by the PLL stability. If it is too large the phase margin of the PLL becomes too low and eventually the PLL becomes unstable. The higher the value of N, the lower the PLL bandwidth. Low phase margin, in addition to impairing PLL stability, leads to jitter peaking in the PLL.
- A phase locked loop with a charge pump phase detector has a low limit for the phase detector sampling frequency (fR). The phase detector sampling frequency should be at least 10 times higher than the PLL open loop unity gain bandwidth.

Both of these factors limit the usable values of N, which makes approximation of the specified output frequency by the PLL more difficult. This leads to the main design trade-off of programmable integer-N PLLs: the specification of  $f_{OUT}$  frequency tolerance should be as wide as possible. If you specify the output frequency very tight and your input and output frequencies are not related as simple integer numbers—for example 25 MHz in, 125 MHz out—you can get a PLL that will have parameters close to the limits of stability. In some cases the output frequency may not be able to be synthesized with the specified accuracy. If you can select the input frequency, choose one that offers a simple multiplication ratio. If you need a 155.52 MHz output, select an integer-related input frequency such as 19.44 MHz or 38.88 MHz.

To illustrate the dependence of PLL parameters on synthesis error, in this example we will generate a 24.576 MHz output from 25 MHz input. We will assume that the maximum VCO frequency of our PLL IC is at least 500 MHz and calculate the N counter value and the phase detector frequency (fR) for synthesis errors of less than 100 ppm of the specified frequency. The synthesis error is:

$$\text{Synthesis Error}_{PPM} = \frac{f_{OUT \text{ SYNTHESIZED}} - f_{OUT \text{ SPECIFIED}}}{f_{OUT \text{ SPECIFIED}}} 10^6$$

## DESIGN CORNER

Synthesis Error, ppm	fR, MHz	N Counter
-86.93	2.2727	173
52.68	1.0870	407
-18.24	0.8065	579

Table 1 - Phase Detector Frequency in Relation to the N Counter

When the synthesis error is -86 ppm, the PLL bandwidth ( $\sim fR/10$ ) could be as wide as 220 kHz. If the error is specified as 20 ppm, the PLL bandwidth cannot be more than about 80 kHz.

You should pick the highest value of synthesis error that your system can tolerate. If the acceptable synthesis error is not specified, the programmer software will try to synthesize the output frequency with minimum error that is achievable from a PLL stability point of view. Note that the synthesis error will be in addition to the accuracy of the input source.

The selection of counter values also affects the PLL's phase noise. A PLL is a low-pass filter for reference frequency phase noise and a high-pass filter for VCO phase noise. Inside the PLL bandwidth, the reference phase noise is multiplied by N and divided by R. The filtered phase noise of the VCO is divided by P at the output. The resulting phase noise at the output is a combination of all those factors. Except when the PLL is used to clean the reference, the contribution of the reference phase noise, even after multiplication, is lower than the VCO phase noise at all frequencies except very close to the carrier (referred to as 'close-in' phase noise). Therefore, to reduce the phase noise at the output, it is advantageous to have as wide a PLL bandwidth as possible. Since VCO phase noise is divided by P, it is preferable to run the VCO at the highest frequency possible. It should be noted that this leads to higher power consumption.

The phase noise describes the noise components of the signal in the frequency domain. The jitter describes the signal noise in the time domain. These parameters are related but their relations are complicated. In case of a PLL, the situation is even more complicated than in case of a direct crystal oscillator since, in addition to random jitter that is related to the random phase noise, the PLL generates deterministic jitter that is related to crosstalk between the PLL building blocks: VCO, phase detector, and the counters. The random part of the jitter follows the same rule as the phase noise—the wider the PLL bandwidth and the higher the VCO frequency, the better random jitter. Deterministic jitter is much more difficult to characterize. Some of it is due to the VCO control voltage (Vc) ripple at fR frequency that passes through the loop filter. This component is attenuated when

(continued on page 29)

$f_R$  is much higher than the PLL bandwidth, so to reduce the deterministic jitter the PLL bandwidth should be low comparing to  $f_R$ . Another rule of thumb is the more dividers are involved, the more crosstalk-related deterministic jitter you will have. One of the consequences of this rule is the more PLLs are used on the chip, the more crosstalk between PLLs. From the deterministic jitter point of view, a single PLL IC will have less jitter than a multiple PLL IC. If your application is sensitive to jitter, consider using a separate PLL IC for each frequency or for the most sensitive frequencies.

Many PLL ICs have a crystal oscillator as part of the IC. These ICs usually have a reference output that is the buffered output of the crystal oscillator. Note that the reference output of a PLL IC most likely will have higher jitter due to the crosstalk from the PLL(s) operating on the same die than a stand-alone crystal oscillator. This additional jitter will be mostly deterministic in nature.

Besides other factors affecting the jitter performance, the PLL jitter is highly dependent on the R, N, and P counter selections. Therefore, the performance of a PLL IC depends on optimal selection of counter values. The jitter performance of a particular input/output frequency combination may vary from the published typical numbers. You should ask for the jitter performance of the requested samples or study the conditions (PLL configuration) in which the published jitter numbers were measured.

Since deterministic jitter depends on crosstalk, it depends on the phase relations between various intermediate frequencies existing inside the IC. Since some of these phase relations can vary from one power-up of the chip to another, the jitter can be different from one power-up of the same chip to another. The jitter can also be different if measured between the rising or the falling edges of the output waveform.

The deterministic jitter of a PLL has some properties that can be used to analyze its impact on the target system. As its name implies, it depends on the finite number of frequencies that are known or could be calculated by considering all possible mixing products of frequencies present in the PLL. The frequencies contributing to deterministic jitter can be measured by a spectrum analyzer. The peak-to-peak value of deterministic jitter is a bounded number that does not grow with the number of period measurements as a peak-to-peak random jitter value does. You should evaluate the target system in terms of how the frequencies causing the deterministic jitter will affect it. The deterministic jitter may create 'multimodal distribution' on the period jitter histogram. However, the multimodal distribution caused by deterministic jitter on the period jitter histograms should not prevent you from using a PLL. For example, the intermediate frequencies resulting in the deterministic jitter could be high enough to be attenuated by the internal PLL(s) of the target system.

## DESIGN CORNER

Some PLL ICs have programmable output buffers. They allow changing the buffer strength vs. output frequency, output level, and buffer supply voltage. The rule of thumb is to use the lowest drive level your application can accept. Too high a drive level could lead to excessive EMI from the board. It can also increase crosstalk. Usually, the high drive is recommended for frequencies above 100 MHz or if the buffer supply voltage is low, for example, 1.8V.

High drive can improve the measured jitter because it reduces conversion of AM to PM at the input of the measurement equipment due to decrease of the rise and fall times.

The layout of the board with a PLL IC requires special attention to power supply pin decoupling. It was mentioned above that the crosstalk is a major contributor to the PLL jitter. Some of the crosstalk comes from inside the IC and cannot be changed. But some of the crosstalk comes from the PLL block's interaction through the power lines. All the power pins must be decoupled with capacitors placed as close to the power pins as possible. When selecting capacitor values for decoupling, remember that a PLL is a multi-frequency system and—together with 100 MHz at the output—it can have frequencies in kHz range on chip that could modulate power supply voltage and VCO frequency and create crosstalk. If you use inductors or ferrite beads for decoupling, look for the possibility of resonances in the power supply lines. PLLs are more sensitive to decoupling than oscillator ICs and can be unstable in conditions where an oscillator IC would be stable.

As mentioned above, the programmable PLLs keep information in OTP and configure themselves during power up. They start in the unprogrammed state. The transition from unprogrammed state to the programmed state could be quite complicated when using multi-PLL ICs with several programmable banks. Because of this, the requirement of the power supply ramp for programmable PLLs is more stringent than in case of ICs with a fixed structure, such as oscillator ICs or simple pin-programmable PLLs. Usually the ramp must rise monotonically without dropouts, and there could be minimum and maximum ramp time requirements.

By following these simple design guidelines, you can use programmable PLLs to replace multiple crystals, crystal oscillators and other reference sources to simplify clock generation in a number of applications. To learn more about our products and the ClockWorks Configurator, visit our [Clock Generation page](#). For additional information about PLL design, we recommend the following resources:

F.M. Gardner, *Phaselock Techniques*, 3rd ed, John Wiley, 2005.  
Keliu Shu, Edgar Sánchez-Sinencio, *CMOS PLL Synthesizers: Analysis and Design*, Springer, 2005.

## Smarter Programming

Version 7 of Flowcode Graphical Programming Environment Offers More Features and Easy Programming for Users of Microchip Hardware

Contributed by Matrix TSL

For more than 40 years, microcontrollers (MCUs) have been used in a vast number of products and industrial applications. In the last several years, MCUs have become so commoditized that the market is awash with them. From 8-bit devices used in off-the-shelf, community-driven Arduino® products to rugged and powerful 32-bit devices, these MCUs are getting into the hands of even novices, who are becoming ‘techies’ practically overnight.

However, while the maker movement has attracted many new and enthusiastic users, a large number of them struggle with developing in C code. That’s where **Flowcode** comes in. As a graphical programming software tool, Flowcode allows users to develop complex electronic and electromechanical systems

quickly and easily. Even those with little experience can develop projects in minutes. Professionals, students, and individuals in the growing maker market can get started with a large suite of supported devices including Microchip’s PIC® MCUs, dsPIC® Digital Signal Controllers (DSCs), as well as the famous Atmel® AVR® devices used in Arduino platforms. It’s no wonder that Flowcode is selling more copies than ever before.

### The Next Generation: Flowcode 7

In order to keep up with the growing user base, Matrix TSL just launched version 7 of Flowcode, unveiling a modern, new user interface (UI) as well as many improvements and features, making it the most user-friendly version to date. While Flowcode 7—like its predecessors—will program Microchip’s 8- and 16-bit PIC MCUs, a significant addition to this latest version is support for the powerful PIC32 series of 32-bit MCUs, which offer incredible processing power for little extra cost. And don’t fret; all programs written in previous versions of Flowcode will directly transfer to this new platform, so you can leverage 32-bit processing power for mathematical functions like speech generation. Flowcode programs will also seamlessly transfer between microcontroller types, allowing you to take advantage of many hardware platforms.



(continued on page 31)

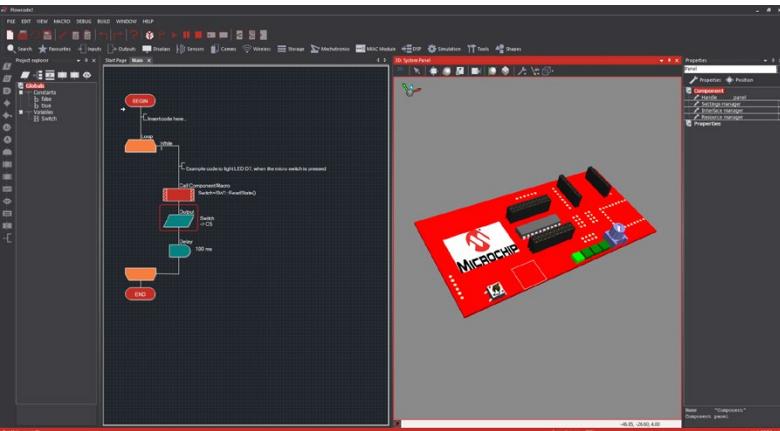
# On the Hardware Side

**E-blocks** are Matrix's small circuit-board family used for rapid prototyping of complex electronic systems. While these boards are typically more expensive than hardware from companies such as Microchip, they are popular with educators and students as well as professional engineers. E-blocks boards incorporate time-saving debug and instrumentation circuitry known as **Ghost technology**, a real-time log of the status of all the pins on the MCU (while a Flowcode program is running). Ghost includes a data recorder, oscilloscope, logic analyzer, packet decoder, and in-circuit/in-system debug. Although developers might obtain their hardware and software elsewhere for free, the time spent developing a project can still be costly. When used together, Flowcode 7 and Matrix's hardware solutions enable users to reduce their development time.

## Flowcode Templates

Developers can save even more time by using Flowcode's template files. Ideal for new users, they feature 3D graphical representations of the development boards, complete with working components such as switches, LEDs, potentiometers, and the like. In addition to looking and feeling like the real hardware, template files also come preconfigured for downloading to the hardware, so any registers, compiler, programmer settings and other options are already set. Each template file includes a very simple example that you can test before writing your own code. You can have your first microcontroller program working in just minutes.

Working hand in hand with Microchip, we have chosen some specific development boards which are available as templates in Flowcode 7. One of these is the **Curiosity Development Board** (DM164137), a cost-effective, fully integrated 8-bit development platform targeted at first-time users, makers and anyone seeking a feature-rich rapid prototyping board. A (beta) screen capture of the Curiosity board as it appears in Flowcode version 7 is shown below.



Curiosity Development Board template in Flowcode 7.

## DESIGN CORNER

The **MPLAB® Xpress Evaluation Board** (DM164140) and a number of other Microchip development boards are in the list of featured templates. Future supported boards are likely to include a 32-bit PIC MCU, such as the **chipKIT™ uc32 Development Board** (TDGL017), which has the same form factor as the Arduino Uno and is compatible with Arduino shields.



MPLAB® Xpress  
Evaluation Board  
(DM164140)



chipKIT™ uc32  
Development Board  
(TDGL017)

These templates can be accessed in the free version of Flowcode 7. While this version offers limited features, components and target devices, it is excellent for learning programming, developing applications at home, or for prototyping designs. Note that the free version of Flowcode 7 cannot be licensed for commercial or educational institution use.

## Testing and Debugging Designs

A number of features within Flowcode 7 allow you to test and debug systems more easily and extensively than when developing projects using text-based code and other languages. Simulation is a core function, allowing you to instantly run your program without having to compile or download it to confirm that it is working correctly.



Simulation of car seat motor and actuator movement being controlled by MCU.

The simulation debugger window becomes active during simulation, allowing you to monitor the entire process as the microcontroller runs the program. You can view

(continued on page 32)

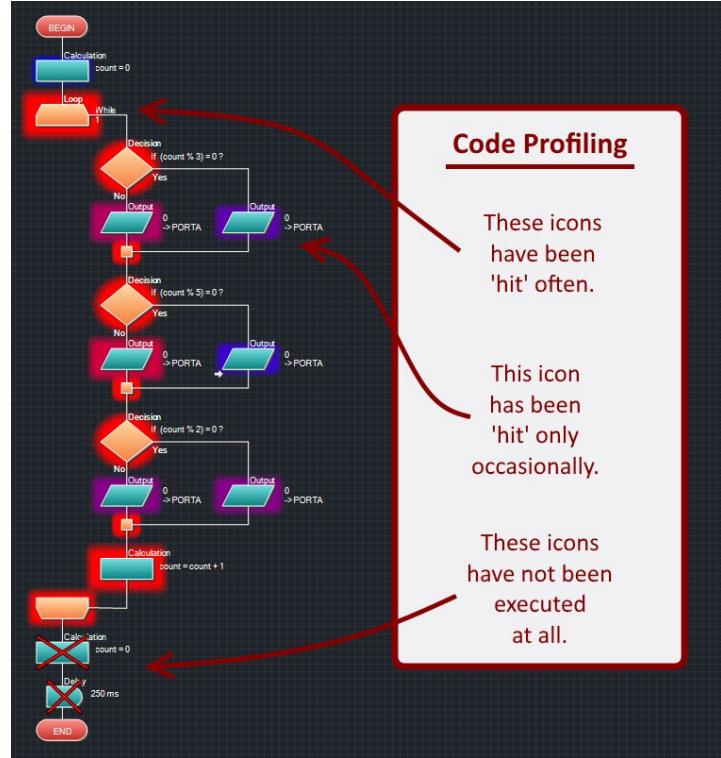
both the values of the variables in your program and the macros being called, and you can change the values of the variables while the simulation is paused, allowing you to perform specific case testing.

Code profiling is a new feature that has been launched for the first time in Flowcode 7. This feature shows when certain steps (shown as icons) have been “hit” during a simulation run, highlighting sections of code that are executed often and potentially redundant, which gives you insight into code that may need optimization for improving program efficiency.

## MPLAB XC Compilers

We have also been working with Microchip to include the MPLAB XC compilers in Flowcode 7, effectively improving the speed at which 8-bit PIC devices compile by more than ten times compared to previous versions of Flowcode. The use of the MPLAB XC compilers also enables Flowcode to compile code for Microchip’s 16- and 32-bit devices.

Visit [Matrix TSL’s website](#) to learn more. From there, you can [download](#) the simulation-only free version of Flowcode 7 or [purchase](#) the appropriate Flowcode license for your needs. 



Code profiling is a new feature available in Flowcode 7.

## Industry's Most Complete Configurable MEMS Timing Solutions

- Two weeks production time
- 5x better vibration tolerance, 500x better mechanical shock tolerance
- High stability, low power, ultra-small packages



# Measuring Time

## RTOS 101: Software Performance Analysis with Tracealyzer

Contributed by Percepio AB

When developing firmware that uses a Real-Time Operating System (RTOS), you may wonder how you can accurately measure your software's performance. The most obvious method that may come to mind is to measure response time, which you'd calculate by using a standard clock measurement of the time it takes to go from point A to point B in the code. The method used to make the calculation would vary, but you could, for example, toggle an I/O pin and take the measurement with a logic analyzer. However, this basic measurement provides only the response time without any insight into which events could have affected it. When using an RTOS, interrupts and other high-priority tasks are likely to interfere with the software's performance, as discussed in the **"Visualize Your RTOS Trace"** article published in an earlier issue of *MicroSolutions*.

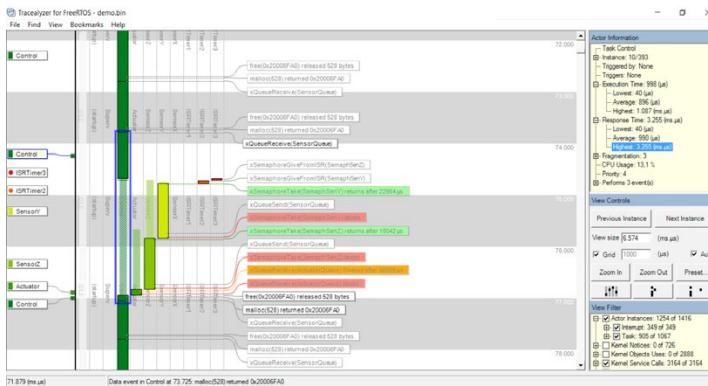


Figure 1 - Tracealyzer Main View

An alternate method to determine software performance is to measure the amount of processor time used by each task, thread or function. For this approach, you could, for example, sample the program counter to produce a high-level overview of the main users of processor cycles. This solution is, in fact, used by several common IDEs. In addition, most ARM-based microcontrollers (MCUs) also provide hardware support for PC sampling. However, this method provides only an average measurement of the typical distribution, which is often inaccurate for less frequent functions/tasks. Rare cases of unusually long execution times—that might actually cause timing problems—are overlooked using such tools.

**Tracealyzer**, on the other hand, provides an exact picture of the run-time behavior at the RTOS level, including many performance statistics like execution time and response time, as well as detailed trace views that explain events that have affected the timing. Such detail gives you valuable insight into the run-time world and facilitates the development, validation and debugging of RTOS- and Linux®-based embedded software.

Tracing an operating system is certainly not a new concept in itself. While tools that visualize task scheduling have been around for many years, they have typically been specific to some operating systems. Moreover, they are often half-hearted utilities that tend to be used as a last resort. Tracealyzer is something else.

(continued on page 34)

Available for several leading operating systems, Tracealyzer provides an innovative visualization system, refined for more than a decade, to offer an unprecedented level of insight into the runtime world of embedded software. Tracealyzer's main view (Figure 1) visualizes task scheduling and other events on a vertical timeline, and includes a rendering of the task scheduling on the left side. The vertical orientation allows other events, such as RTOS calls or custom User Events, to be displayed in horizontal text labels on the right side.

The rectangles in the scheduling trace correspond to intervals of uninterrupted execution, called "fragments" in Tracealyzer. The term "actor" is used to denote all execution contexts in the traced system; it includes things like tasks and interrupt handlers. Task scheduling can be rendered in different ways by using the buttons under the Zoom area to select different view modes. In the view shown in Figure 1, the fragments are ordered in multiple columns, one for each actor.

Something else unique to Tracealyzer is its display of statistics and visualizations based on instance metrics. An "instance" is a specific execution of an actor from the time it is activated until it is suspended. In the case of a cyclical task, each task instance corresponds to one iteration of the task's main loop. As you can see, this instance concept is central to Tracealyzer. When an actor fragment in the Tracealyzer main view is clicked, the Actor Instance is highlighted with a blue rectangle, as shown in Figure 1. Performance metrics like execution time and response time are calculated for each instance. These metrics can be visualized either via detailed plots over time or in higher-level histograms that show the distribution. The latter is displayed in Figure 2, which shows that during a sample trace, the response time of a control task is a maximum of 3255  $\mu$ s, while the maximum execution time is just 1087  $\mu$ s. The histogram bars are linked to a search function, which allows you to find the corresponding locations in the main trace view and see the cause of any anomalies.

You might now wonder how the stream of task scheduling events is grouped into task instances. This is a good question because grouping is not obvious for cyclical RTOS tasks, where an instance corresponds to a single iteration of the main loop.

## DESIGN CORNER

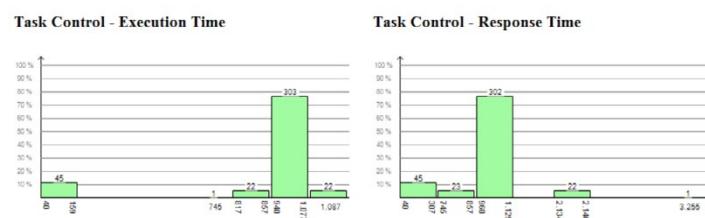


Figure 2 - Tracealyzer Sample Histograms

In such cases, an instance is identified by a blocking RTOS call—such as a "Delay" call—that is placed somewhere in the loop. Now you might wonder, since a task might perform multiple calls, how does Tracealyzer know where to end the current instance and begin a new instance? To resolve this conundrum, "Instance Finish Events" (IFEs) are used. By default, standard rules specify which RTOS calls should be counted as IFEs: "Delay" calls and "ReceivefromQueue" calls. Using this default requires no extra configuration and is usually correct. However, when these rules are not suitable, you can generate IFEs that indicate that the instance is complete by calling a certain function in the Tracealyzer recorder library. Now you can manually group the events into instances, thereby customizing the meaning of the timing statistics. For example, in Figure 1, the dark green control task is divided into multiple instances even though no task switches occur at these points.

In the end, Tracealyzer is a one-of-a-kind software tool for accurately measuring software performance. To learn how Tracealyzer can help you discover timing problems with your RTOS-based firmware [download a free trial](#). Tracealyzer will help you understand, troubleshoot and optimize your software.

FreeRTOS+Trace for PIC32 is a unique version of Tracealyzer that is specific to PIC32 MCUs. Order the Standard Version (TSW1009) or the Professional Version (TSW1010) at unheard of prices from [microchipDIRECT](#). 

# Scalability, Meet Flexibility

MPLAB® X Integrated Development Environment



MPLAB® X Integrated Development Environment (IDE) provides embedded design and debug for all of Microchip's PIC® microcontrollers on three operating systems (Windows®, MAC OS® and Linux®).

## MPLAB X IDE Features

- Provides Call Graph for navigating complex code
- Supports multiple configurations within your projects
- Supports multiple versions of the same compiler
- Support for multiple debug tools of any type
- Supports live parsing
- Import MPLAB 8 IDE projects
- Supports hyperlinks
- Supports live code templates
- Supports the ability to enter file code templates
- Track changes within your own system (history)
- Can reformat to your own code format style



[www.microchip.com/mplabx](http://www.microchip.com/mplabx)