

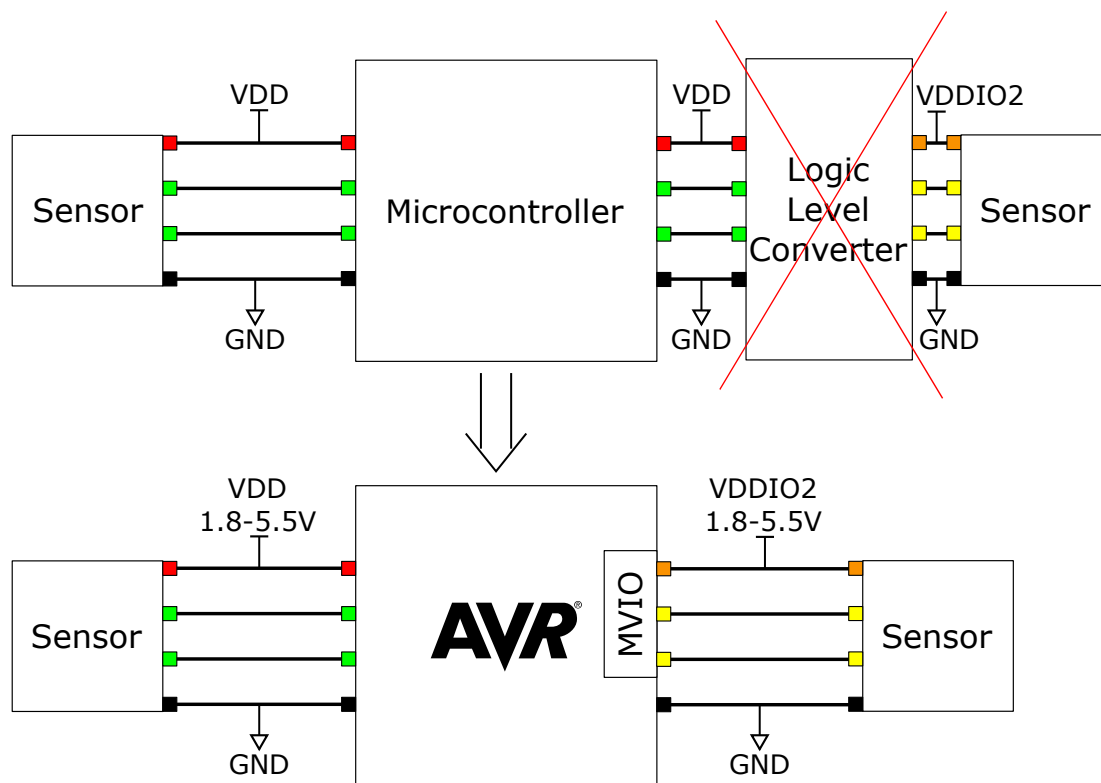
Getting Started with Multi-Voltage I/O (MVIO)

Introduction

Authors: Johan Vaarlid, Microchip Technology Inc.

The AVR® DB Family of microcontrollers is equipped with a configurable integrated level shifter. The Multi-Voltage I/O (MVIO) feature allows a subset of the I/O pins to be powered by a different I/O voltage domain VDDIO2. The rest of the I/O pins runs on VDD. Both VDD and VDDIO2 have the same voltage range. Consult the *Electrical Characteristics* section of the data sheet for the exact voltage range. This eliminates the need of having external level shifters for communication or control of external components running on a different voltage level. Eliminating external logic level shifters will, in turn, reduce the BOM and free up PCB real estate.

Figure 1. Multi-Voltage I/O can Completely Replace External Logic Level Converters



This technical brief describes how the Multi-Voltage I/O feature works on the AVR DB Family of microcontrollers. It covers the following use cases:

- **Polling VDDIO2 Status:**
Checking if the VDDIO2 voltage is within the acceptable range by reading the Status bit.
- **Interrupt on VDDIO2 Status:**
Make use of the interrupt that occurs when the Status bit changes state.
- **Measuring VDDIO2:**

Use the internal ADC to measure the VDDIO2 voltage.

- **Blink a 3V LED with VDD at 1.8V:**

A basic example of MVIO blinking an LED with a forward voltage higher than VDD.

Note: The code examples were developed on AVR128DB48 Curiosity Nano (EV35L43A), and are available on GitHub.



[View Code Example on GitHub](#)

Click to browse repository

Table of Contents

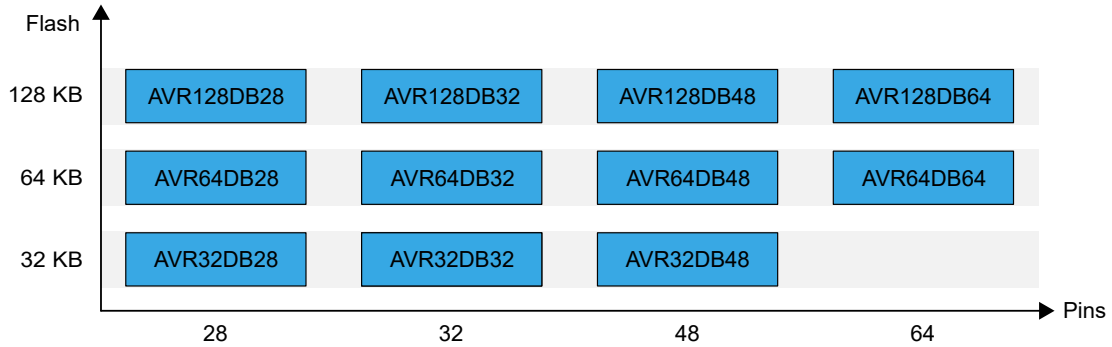
Introduction.....	1
1. Relevant Devices.....	4
2. Overview.....	5
3. Hardware Configuration.....	7
3.1. Using the Curiosity Nano AVR128DB48.....	7
3.2. Supplying Power To VDDIO2.....	7
4. Reading the Status Bit.....	9
5. Interrupt on VDDIO2 Status	10
6. Measuring VDDIO2.....	11
7. Blinking a 3V LED with VDD at 1.8V.....	15
8. References.....	17
9. Revision History.....	18
10. Appendix.....	19
The Microchip Website.....	22
Product Change Notification Service.....	22
Customer Support.....	22
Microchip Devices Code Protection Feature.....	22
Legal Notice.....	23
Trademarks.....	23
Quality Management System.....	24
Worldwide Sales and Service.....	25

1. Relevant Devices

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

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

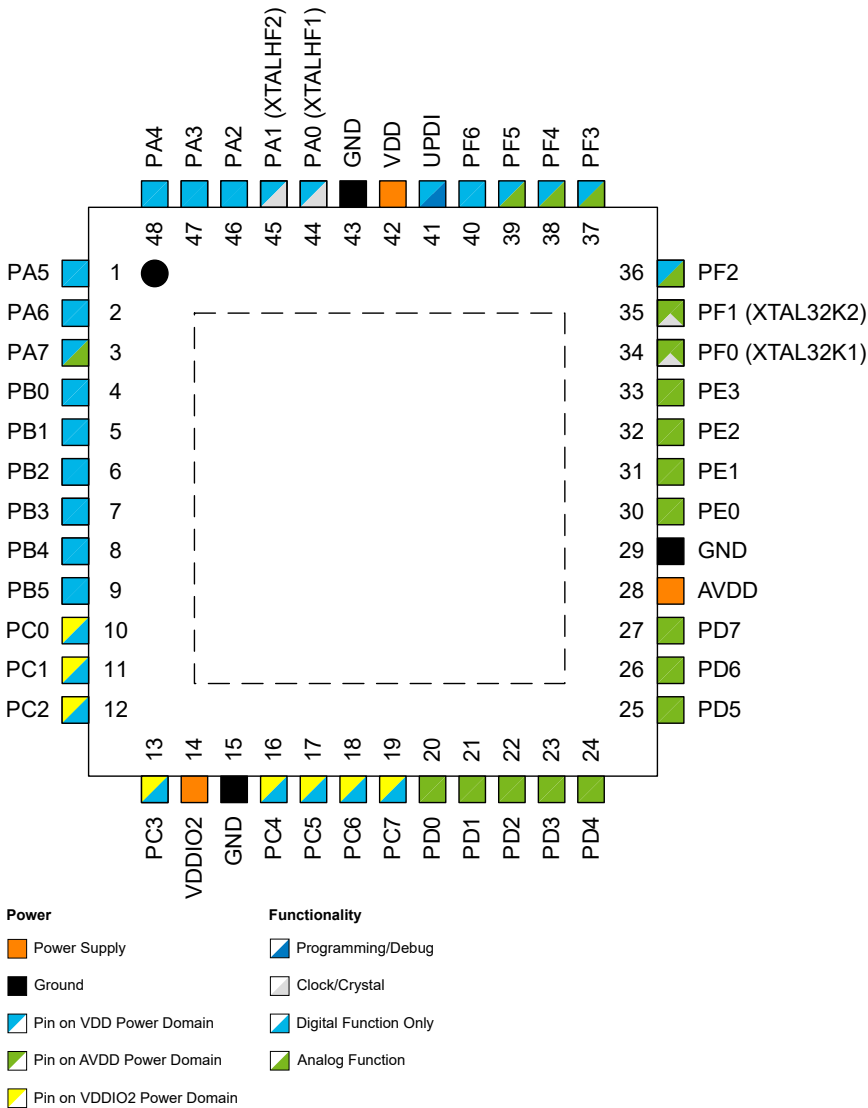
Figure 1-1. AVR® DB Family Overview



2. Overview

The MPIO pins, located on PORTC, are powered by a separate VDDIO2 pin while the rest are powered by VDD. This allows for additional power domain at a different voltage level within the limits of the Electrical Characteristics of each device.

Figure 2-1. 48-Pin VQFN/TQFP Pinout Diagram

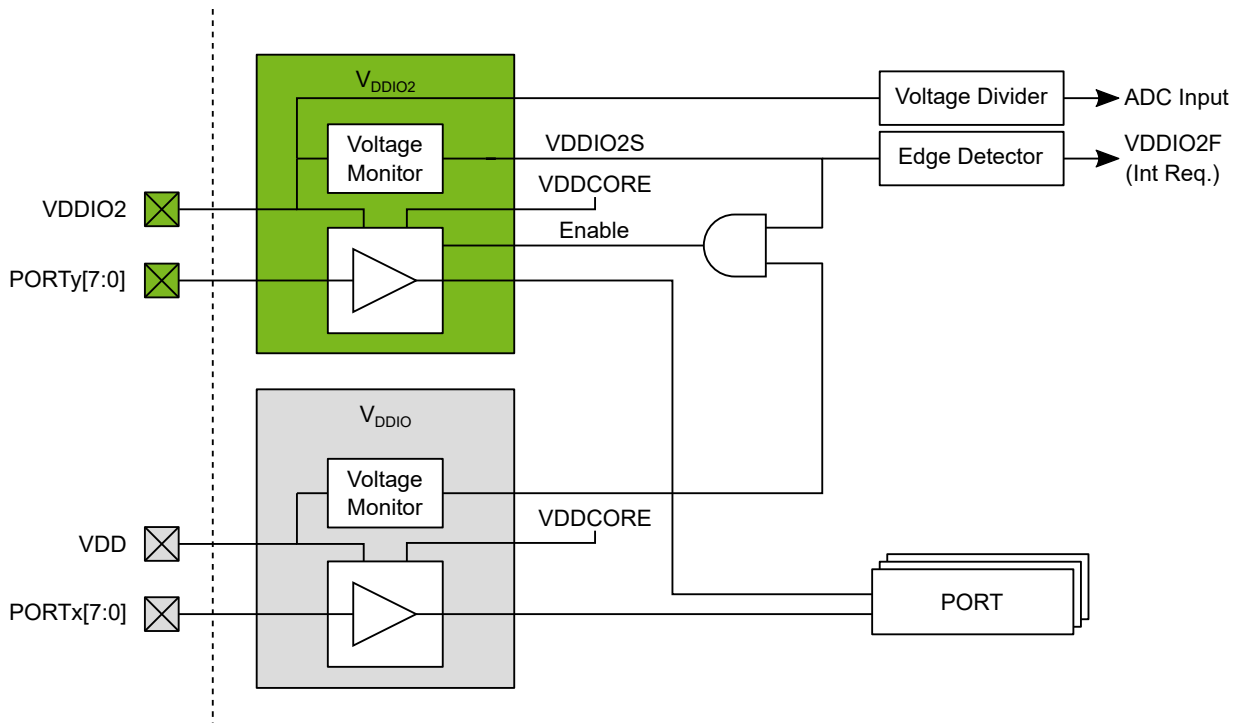


These pins are capable of the same digital behavior as regular I/O pins, for example GPIO, serial communication (USART, SPI, I²C), or connected to PWM peripherals. The input Schmitt Trigger levels are scaled according to the VDDIO2 voltage, as described in the Electrical Characteristics section of the data sheet.

A configuration fuse determines the MPIO supply mode. The loss or gain of power on VDDIO2 is signaled by a status register bit. This status bit has corresponding interrupt and event functionality.

A divided-down VDDIO2 voltage is available as input to the ADC.

Figure 2-2. MVIO Block Diagram



3. Hardware Configuration

3.1 Using the Curiosity Nano AVR128DB48

To illustrate the capabilities of Multi-Voltage I/O an AVR128DB48 Curiosity Nano is used. Schematics of the Curiosity Nano PCB can be found in the [References](#) section. To enable MVIO on this development board, perform the below steps:

1. Remove the resistor right below the VDDIO2 label to disconnect VDDIO2 from VDD. This is resistor R204 in [Figure 3-2](#).
2. Connect VDDIO2 + to 3.3V of a second power voltage source.
3. Connect VDDIO2 - to GND of a second voltage source.

Refer to the [AVR128DB48 Curiosity Nano User Guide](#) for a guide on how to reverse this process.

Any voltage between 1.8V and 5.5V may be applied to VDDIO2, but for the following examples, we will run at 3.3V.

Figure 3-1. AVR128DB48 Curiosity Nano MVIO

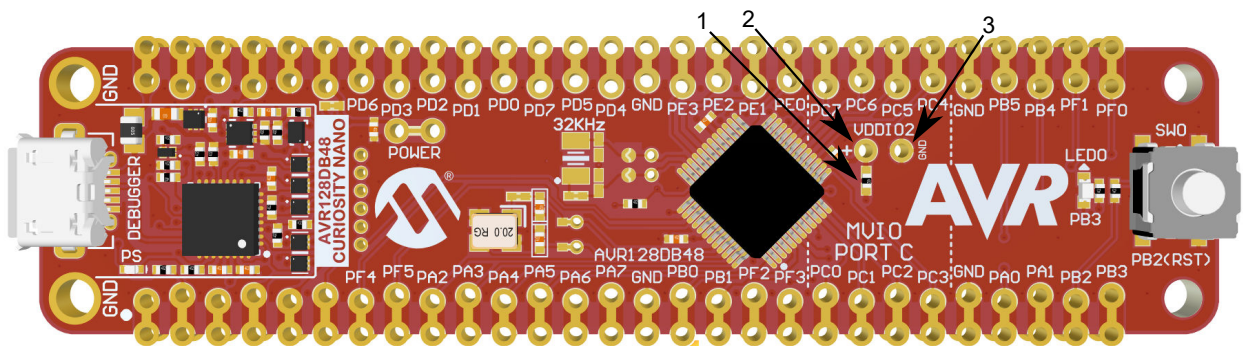
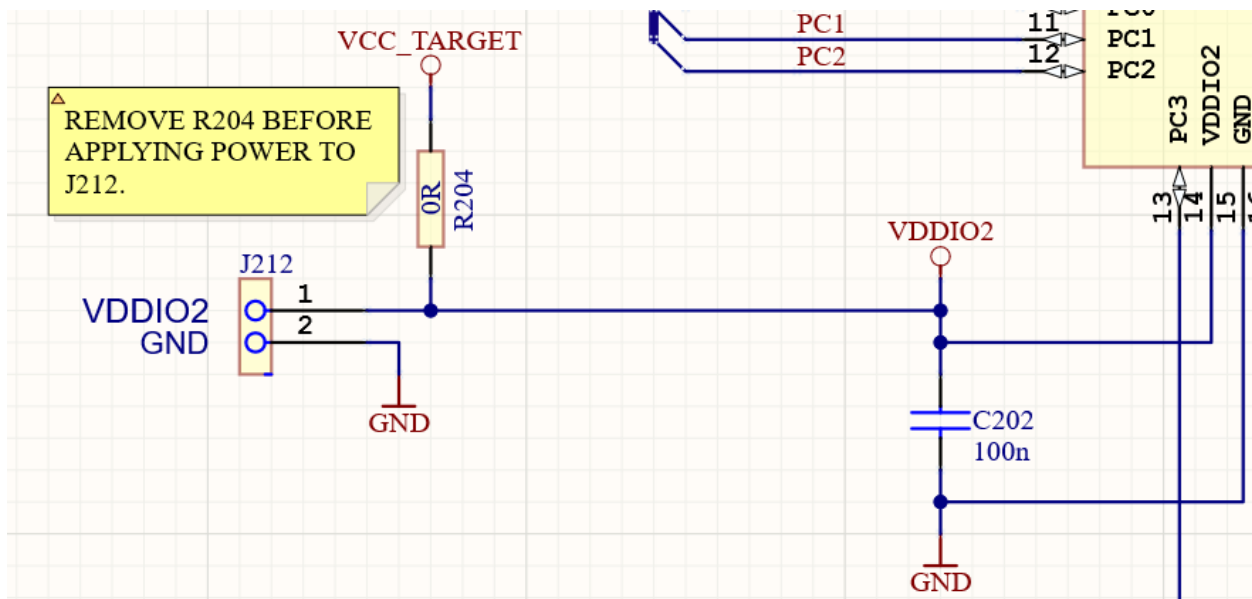


Figure 3-2. Snippet From the AVR128DB48 Curiosity Nano Schematic



3.2 Supplying Power To VDDIO2

The Multi-Voltage I/O (MVIO) can be configured in one of two supply modes:

- Single-Supply mode, where the MVIO-capable I/O pins are powered at the same voltage level as the non-MVIO capable pins, i.e., VDD. The user must connect the VDDIO2 pin(s) to the VDD pin(s).
- Dual-Supply mode, where the MVIO-capable I/O pins are supplied by the VDDIO2 voltage, which may be different from the voltage supplied to the VDD pin(s)

Figure 3-3. FUSE.SYSCFG1 - Enabling Dual Supply Mode

8.8.2.5 System Configuration 1

Name: SYSCFG1
Offset: 0x06
Default: 0x08
Property: -

The default value given in this fuse description is the factory-programmed value, and should not be mistaken for the Reset value.

Bit	7	6	5	4	3	2	1	0
				MVSYS_CFG[1:0]			SUT[2:0]	
Access				R	R	R	R	R
Default				0	1	0	0	0

Bits 4:3 – MVSYS_CFG[1:0]

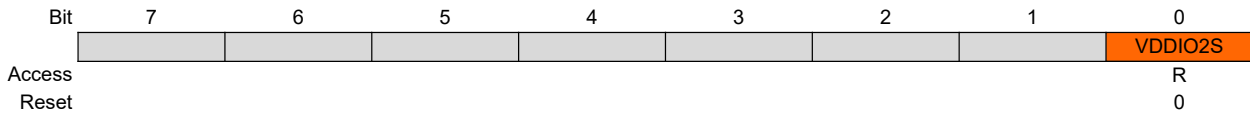
This bit field controls the power supply mode.

Value	Name	Description
0x00	Reserved	-
0x01	DUAL	Device used in a dual supply configuration.
0x02	SINGLE	Device used in a single supply configuration.
0x03	Reserved	-

4. Reading the Status Bit

Before using the MVIO feature, it is recommended to check the VDDIO2 Status (VDDIO2S) bit in the MVIO.STATUS (Status) register, to make sure that the voltage level of VDDIO2 is within the acceptable range of operation. If this bit is '1', then VDDIO2 is within the acceptable range of operation.

Figure 4-1. MVIO.STATUS - Reading the Status bit



Bit 0 – VDDIO2S VDDIO2 Status

This bit shows the state of the VDDIO2 voltage level.

Writing to this bit has no effect.

Value	Description
0	The VDDIO2 supply voltage is below the acceptable range for operation. The MVIO pins are tri-stated.
1	The VDDIO2 supply voltage is within the acceptable range for operation. The MVIO pin configurations are loaded from the corresponding PORT registers.

```

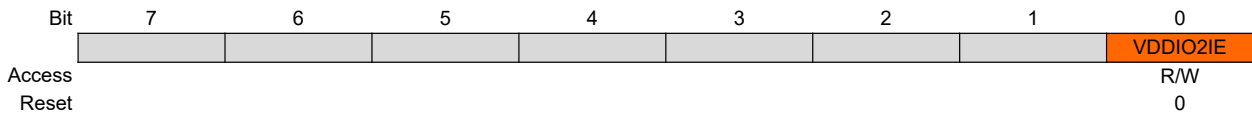
//Polling the VDDIO2S bit
if(MVIO.STATUS & MVIO_VDDIO2S_bm)
{
    //Do something if VDDIO2 is within acceptable range.
}

```

5. Interrupt on VDDIO2 Status

When the state of the VDDIO2 Status bit changes, the VDDIO2 interrupt flag will be raised. To be able to catch this interrupt immediately, it is necessary to enable both the VDDIO2 interrupt and global interrupts in general.

Figure 5-1. MVIO.INTCTRL - Enable the VDDIO2 Interrupt



Bit 0 – VDDIO2IE VDDIO2 Interrupt Enable

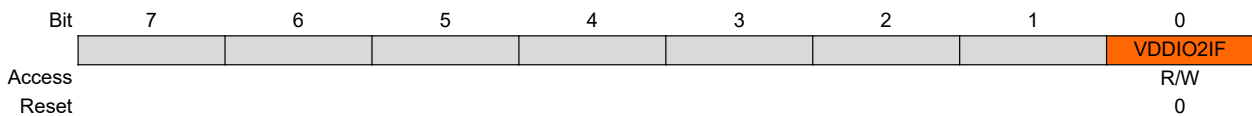
This bit controls whether the interrupt for a VDDIO2 state change is enabled or not.

Value	Description
0	The VDDIO2 interrupt is disabled
1	The VDDIO2 interrupt is enabled

```
/*Enable the VDDIO2 interrupt*/
MVIO.INTCTRL |= MVIO_VDDIO2IE_bm;
/*Enable Global interrupts*/
sei();
```

The MVIO vector will be called when the interrupt is triggered. Before returning from the Interrupt Service Routine (ISR), the VDDIO2 (VDDIO2IF) interrupt flag in the MVIO Interrupt Flags (MVIO.INTFLAGS) register must be cleared. This is done by writing a '1' to the flag.

Figure 5-2. MVIO.INTFLAGS - Clear VDDIO2 Interrupt Flag



Bit 0 – VDDIO2IF VDDIO2 Interrupt Flag

This flag is cleared by writing a '1' to it.

This flag is set when the VDDIO2 Status (VDDIO2S) bit in MVIO.STATUS changes value.

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the VDDIO2 Interrupt Flag.

```
ISR(MVIO_MVIO_vect)
{
    /* The VDDIO2 interrupt has been triggered */
    LED0_toggle();
    /* Clear the CFD interrupt flag */
    MVIO.INTFLAGS |= MVIO_VDDIO2IF_bm;
}
```

6. Measuring VDDIO2

The internal ADC has VDDIO2 divided by 10 as an input, making it possible to measure the voltage of the VDDIO2 domain even with the internal 1.024V reference.

First, we select the 1.024V reference for ADC0 in the ADC0REF register, as shown below. Note that this register is located in the Voltage Reference Section(VREF), not the ADC section.

Figure 6-1. VREF.ADC0REF - Select 1V024 as Reference for ADC0

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

Bits 2:0 – REFSEL[2:0] Reference Select

This bit field controls the reference voltage level for ADC0.

Note:

- The values given for internal references are only typical. Refer to the *Electrical Characteristics* section for further details.

Value	Name	Description
0x0	1V024	Internal 1.024V reference ⁽¹⁾
0x1	2V048	Internal 2.048V reference ⁽¹⁾
0x2	4V096	Internal 4.096V reference ⁽¹⁾
0x3	2V500	Internal 2.500V reference ⁽¹⁾
0x4	-	Reserved
0x5	VDD	VDD as reference
0x6	VREFA	External reference from the VREFA pin
0x7	-	Reserved

```
/*Select the 1.024V referance for ADC0*/
VREF.ADC0REF = VREF_REFSEL_1V024_gc;
```

Select the VDDIO2DIV10 as input for ADC0 on the positive input pin.

Figure 6-2. ADC0.MUXPOS - Select VDDIO2DIV10 as Input for ADC0

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

MUXPOS[6:0]

Bits 6:0 – MUXPOS[6:0] MUX Selection for Positive ADC Input

This bit field selects which analog input is connected to the positive input of the ADC. If this bit field is changed during a conversion, the change will not take effect until the conversion is complete.

Value	Name	Description
0x00–0x0F	AIN0-AIN15	ADC input pin 0-15
0x10–0x15	AIN16-AIN21	ADC input pin 16-21
0x16–0x1F	-	Reserved
0x20–0x3F	-	Reserved
0x40	GND	Ground
0x41	-	Reserved
0x42	TEMPSENSE	Temperature sensor
0x44	VDDDIV10	VDD divided by 10
0x45	VDDIO2DIV10	VDDIO2 divided by 10
0x48	DAC0	DAC0
Other	-	Reserved

```
/*Select VDDIO2DIV10 as input for ADC0*/
ADC0.MUXPOS = ADC_MUXPOS_VDDIO2DIV10_gc;
```

The ADC has built-in support for oversampling. This is especially useful when measuring a DC voltage, as it mitigates a lot of the periodic noise.

Figure 6-3. ADC0.CTRLB - Enabling Oversampling with 16 Samples

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

SAMPNUM[2:0]

Bits 2:0 – SAMPNUM[2:0] Sample Accumulation Number Select

This bit field selects how many consecutive ADC sampling results are accumulated automatically. When this bit field is written to a value greater than 0x0, the according number of consecutive ADC sampling results are accumulated into the ADC Result (ADCn.RES) register.

Value	Name	Description
0x0	NONE	No accumulation
0x1	ACC2	2 results accumulated
0x2	ACC4	4 results accumulated
0x3	ACC8	8 results accumulated
0x4	ACC16	16 results accumulated
0x5	ACC32	32 results accumulated
0x6	ACC64	64 results accumulated
0x7	ACC128	128 results accumulated

```
/* Enable 16 oversamples*/
ADC0.CTRLB = ADC_SAMPNUM_ACC16_gc;
```

Enable ADC0.

Figure 6-4. ADC0.CTRLA - Enable ADC0

Bit	7	6	5	4	3	2	1	0
	RUNSTDBY		CONVMODE	LEFTADJ	RESSEL[1:0]		FREERUN	ENABLE
Access	R/W		R/W	R/W	R/W	R/W	R/W	R/W
Reset	0		0	0	0	0	0	0

Bit 0 – ENABLE ADC Enable

Value	Description
0	ADC is disabled
1	ADC is enabled

```
/*Enable ADC0*/
ADC0.CTRLA = ADC_ENABLE_bm;
```

It is now possible to start measurement by writing a '1' to the STCON (Start Conversion) bit in the Command (ADC0.COMMAND) register.

Figure 6-5. ADC0.COMMAND - Start a Measurement

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

Bit 0 – STCONV Start Conversion

Writing a '1' to this bit will start a conversion as soon as any ongoing conversions are completed. If in Free-Running mode, this will start the first conversion. STCONV will read as '1' as long as a conversion is in progress. When the conversion is complete, this bit is automatically cleared. Writing a '0' to this bit has no effect.

```
/*Start an ADC conversion*/
ADC0.COMMAND = ADC_STCONV_bm;
```

After a conversion is started, it is necessary to wait for it to be done before reading the result. The RESRDY (Result Ready) interrupt flag in the Interrupt Flags (ADC0.INTFLAGS) register will be triggered when a result is ready in the Result (ADC0.RES) register.

Figure 6-6. ADC0.INTFLAGS - Check RESRDY Interrupt Flag

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

Bit 1 – WCMP Window Comparator Interrupt Flag

This window comparator flag is set when the measurement is complete and if the result matches the selected Window Comparator mode defined by the WINCM bit field in the Control E (ADCn.CTRLE) register. The comparison is done at the end of the conversion. The flag is cleared by either writing a '1' to the bit position or by reading the Result (ADCn.RES) register. Writing a '0' to this bit has no effect.

Bit 0 – RESRDY Result Ready Interrupt Flag

The Result Ready interrupt flag is set when a measurement is complete and a new result is ready. The flag is cleared by either writing a '1' to the bit location or by reading the Result (ADCn.RES) register. Writing a '0' to this bit has no effect.

```
/*Wait for the ADC conversion to be done*/
while(!(ADC0.INTFLAGS & ADC_RESRDY_bm));
```

The result can be read in the Result (ADC0.RES) register. The result must be divided by 16 due to the oversampling. This is easily done by right shifting the value in ADC0.RES 4 bits. The RESRDY interrupt flag will be automatically cleared when ADC0.RES is read.

Figure 6-7. ADC0.RES - Read the Result

The ADCn.RESL and ADCn.RESH register pair represents the 16-bit value, ADCn.RES. The low byte [7:0] (suffix L) is accessible at the original offset. The high byte [15:8] (suffix H) can be accessed at offset + 0x01.

Bit	15	14	13	12	11	10	9	8
RES[15:8]								
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
Bit	7	6	5	4	3	2	1	0
RES[7:0]								
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:8 – RES[15:8] Result High Byte

This bit field constitutes the high byte of the ADCn.RES register, where the MSb is RES[15].

Bits 7:0 – RES[7:0] Result Low Byte

This bit field constitutes the low byte of the ADCn.RES register.

```
/*Read the RES register and shift the result 4 bits to compensate for the 16 oversamples*/
uint16_t result = (ADC0.RES>>4);
```

7. Blinking a 3V LED with VDD at 1.8V

Following the instructions in the [Hardware Configuration](#) section to supply VDDIO2 with 3.3V and VDD with 1.8V. Make sure that VDDIO2 and VDD have the correct voltage levels by using a multi-meter.

Following the instruction in the [4. Reading the Status Bit](#) section, we check that VDDIO2 is within acceptable range before starting the blinking.

Make sure the LED has a forward voltage of approximately 3V by using a multimeter with diode testing capabilities. A green LED usually has a forward voltage of approximately 3V.

An external green LED in series with a resistor of at least 15 Ohms needs to be connected to PC0, like shown in the figure below.

If a different LED is used, the resistor value needs to change. This is calculated using the following formula where V_{LED} is the voltage drop of the LED and A_{LED} is current through the LED:

$$R = \frac{V_{DDIO2} - V_{LED}}{A_{LED}}$$



The AVR DB can not be programmed below 2V, so to ensure proper operation, set VDD to 3.3V by dragging and dropping the provided text file "VDD3V3.txt" onto the "CURIOSITY" drive on your PC. When programming is done, lower VDD to 1.8V by dragging and dropping "VDD1V8.txt" to the "CURIOSITY" drive. Both .txt files can be found in the GitHub repository linked in the introduction.

Figure 7-1. Curiosity Nano AVR128DB48 - LED Circuit Physical Layout

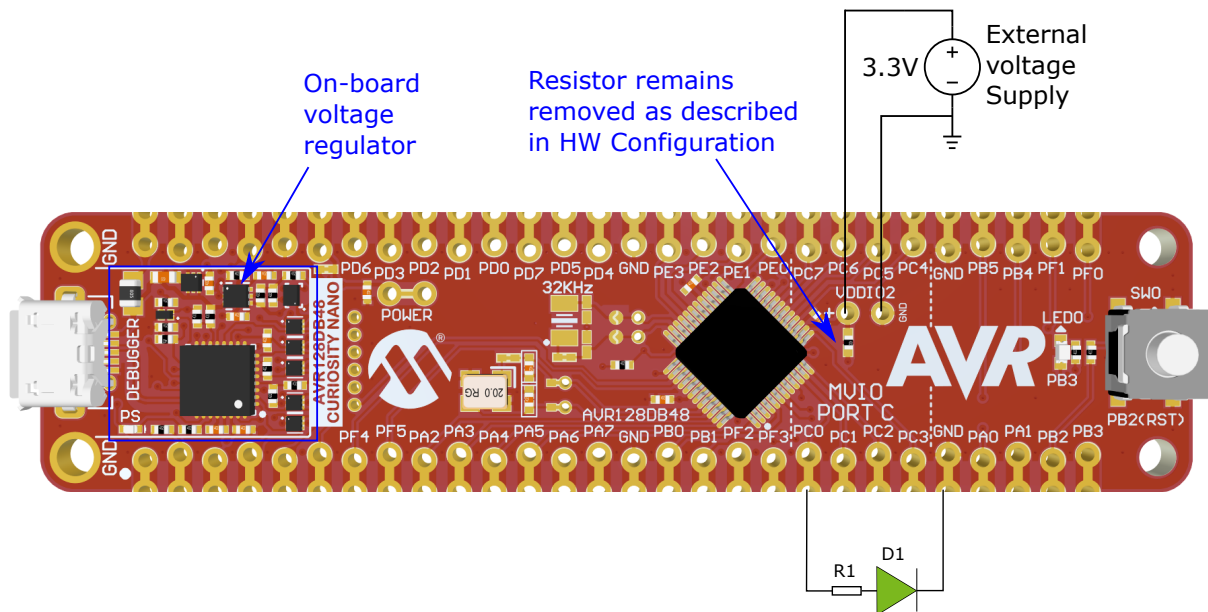
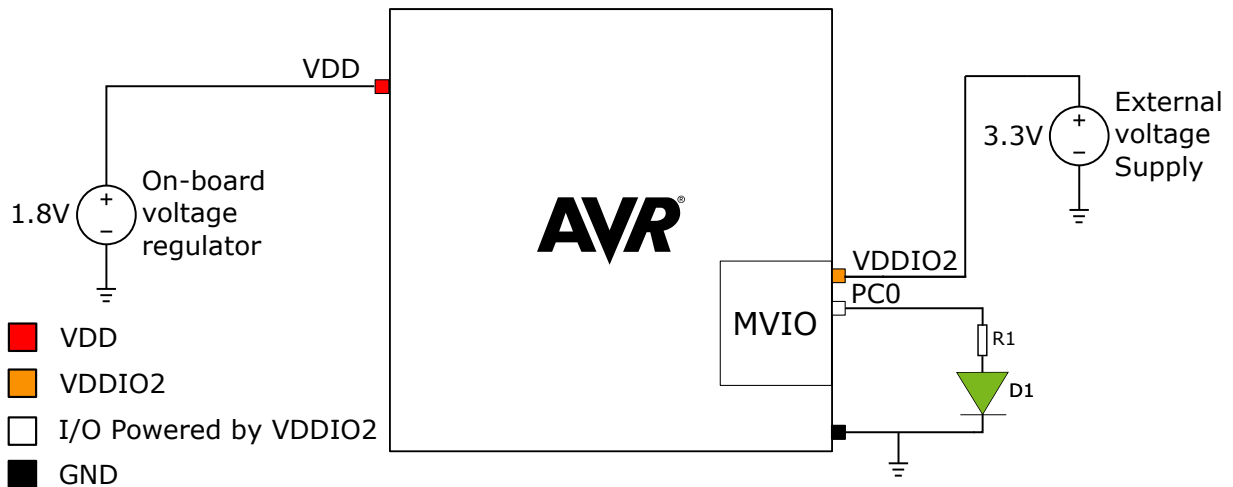


Figure 7-2. LED Circuit Schematics



```

/*Blinking a 3V LED with VDD at 1.8V*/

/*Set LED pin to output*/
void LED_PC0_init(void)
{
    PORTC.DIRSET = PIN0_bm;
}

/*Toggle LED pin*/
void LED_PC0_toggle(void)
{
    PORTC.OUTTGL=PIN0_bm;
}

int main(void)
{
    LED_PC0_init();
    while (1)
    {
        /*Check if VDDIO2 is within acceptable range*/
        if(MVIO.STATUS & MVIO_VDDIO2S_bm)
        {
            /*Blink LED at PC0 forever*/
            while (1)
            {
                LED_PC0_toggle();
                _delay_ms(250);
            }
        }
    }
}

```

The code for this example is available in the **blinking-a-3v0-led-with-vdd-at-1v8** folder in these github repositories



View Code Example on GitHub
Click to browse repository

8. References

1. AVR128DB48 product page: www.microchip.com/wwwproducts/en/AVR128DB48.
2. AVR128DB48: www.microchip.com/DS40002247.
3. Curiosity Nano AVR128DB48 product page: www.microchip.com/DevelopmentTools/ProductDetails/PartNO/EV35L43A.
4. AVR128DB48 Curiosity Nano User Guide: www.microchip.com/DS50003037.
5. Curiosity Nano AVR128DB48 Schematics ww1.microchip.com/downloads/en/DeviceDoc/AVR128DB48_Curiosity_Nano_Schematics.pdf.

9. Revision History

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

10. Appendix

Code examples for each use case in this technical brief.

Example 10-1. Polling VDDIO2 Status

```
#define F_CPU 4000000ul

#include <avr/io.h>

int main(void)
{
    while (1)
    {
        /*Check if VDDIO2 is within acceptable range*/
        if(MVIO.STATUS & MVIO_VDDIO2S_bm)
        {
            /* Replace with your application code */
        }
    }
}
```

Example 10-2. Interrupt on VDDIO2 Status

```
#define F_CPU 4000000ul

#include <avr/io.h>
#include <avr/interrupt.h>

void LED0_init(void) {
    PORTF.DIRSET = PIN5_bm;
}

void LED0_toggle(void) {
    PORTF.OUTTGL = PIN5_bm;
}

int main(void)
{
    /*Enable the VDDIO2 interrupt*/
    MVIO.INTCTRL |= MVIO_VDDIO2IE_bm;

    /*Enable Global interrupts*/
    sei();
    LED0_init();

    /* Replace with your application code */
    while (1)
    {
    }
}

ISR(MVIO_MVIO_vect)
{
    /* The VDDIO2 interrupt has been triggered */
    LED0_toggle();
    /* Clear the CFD interrupt flag */
    MVIO.INTFLAGS |= MVIO_VDDIO2IF_bm;
}
```

Example 10-3. Measuring VDDIO2

```
#define F_CPU 4000000ul
/*Initialize ADC0*/
void adc0_init(void)
{
    /*Select the 1.024V reference for ADC0*/
    VREF.ADC0REF = VREF_REFSEL_1V024_gc;
```

```

/* Enable 16 over samples*/
ADC0.CTRLB = ADC_SAMPNUM_ACC16_gc;

/*Enable ADC0*/
ADC0.CTRLA = ADC_ENABLE_bm;
}

/*Measure VDDIO2 using the internal ADC*/
uint16_t read_VDDIO2()
{
    /*Select VDDIO2DIV10 as input for ADC0*/
    ADC0.MUXPOS = ADC_MUXPOS_VDDIO2DIV10_gc;

    /*Start an ADC conversion*/
    ADC0.COMMAND = ADC_STCONV_bm;

    /*Wait for the ADC conversion to be done*/
    while(!(ADC0.INTFLAGS & ADC_RESRDY_bm));

    /*Read the RES register and shift the result 4 bits to compensate for the
16 oversamples*/
    uint16_t result = (ADC0.RES>>4);

    return result;
}

int main(void)
{
    adc0_init();
    /* Replace with your application code */
    while (1)
    {
        /*Read the VDDIO2 voltage continously*/
        read_VDDIO2();
    }
}

```

Example 10-4. Blink a 3V LED with VDD at 1.8V

```

#define F_CPU 4000000ul

#include <avr/io.h>
#include <util/delay.h>

void LED_PC0_init(void)
{
    PORTC.DIRSET = PIN0_bm;
}

void LED_PC0_toggle(void)
{
    PORTC.OUTTGL=PIN0_bm;
}

int main(void)
{
    LED_PC0_init();
    while (1)
    {
        /*Check if VDDIO2 is within acceptable range*/
        if(MVIO.STATUS & MVIO_VDDIO2S_bm)
        {
            /*Blink LED at PC0 forever*/
            while (1)
            {
                LED_PC0_toggle();
                _delay_ms(250);
            }
        }
    }
}

```



The Microchip Website

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

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

Product Change Notification Service

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

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

Customer Support

Users of Microchip products can receive assistance through several channels:

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

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

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

Microchip Devices Code Protection Feature

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

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

Legal Notice

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

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

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

Trademarks

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

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

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

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

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

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

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

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

ISBN: 978-1-5224-6665-9

Quality Management System

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

Worldwide Sales and Service

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