



This version (06 Dec 2018 15:29) was **approved** by MRaninec.
The [Previously approved version](#) (03 Dec 2018 17:33) is available.

EVAL-CN0429-EBZ Overview

Gas detection instruments are used in a wide range of applications ranging from home air quality measurement devices to industrial solutions for detecting toxic gases. Many of these instruments use electrochemical gas sensors. This sensor technology requires specialized front-end circuitry for biasing and measurement.

By utilizing built-in diagnostics features (such as impedance spectroscopy or bias voltage pulsing and ramping) it is possible to inspect sensor health, compensate for accuracy drift due to aging or temperature, and estimate the remaining lifetime of the sensor right at the edge of the sensor network without user intervention. This functionality allows smart, accurate sensor replacement at the individual edge nodes. An integrated, ultra low power microcontroller directly biases the electrochemical gas sensor and runs onboard diagnostic algorithms.

The [CN0429](#) circuit shows how an electrochemical gas sensor is connected to the potentiostat circuit and how it is biased and measured. Common 2-lead, 3-lead, and 4-lead electrochemical gas sensors can be used interchangeably. The integration of this signal chain dramatically reduces cost, size, complexity, and power consumption at the sensor node.

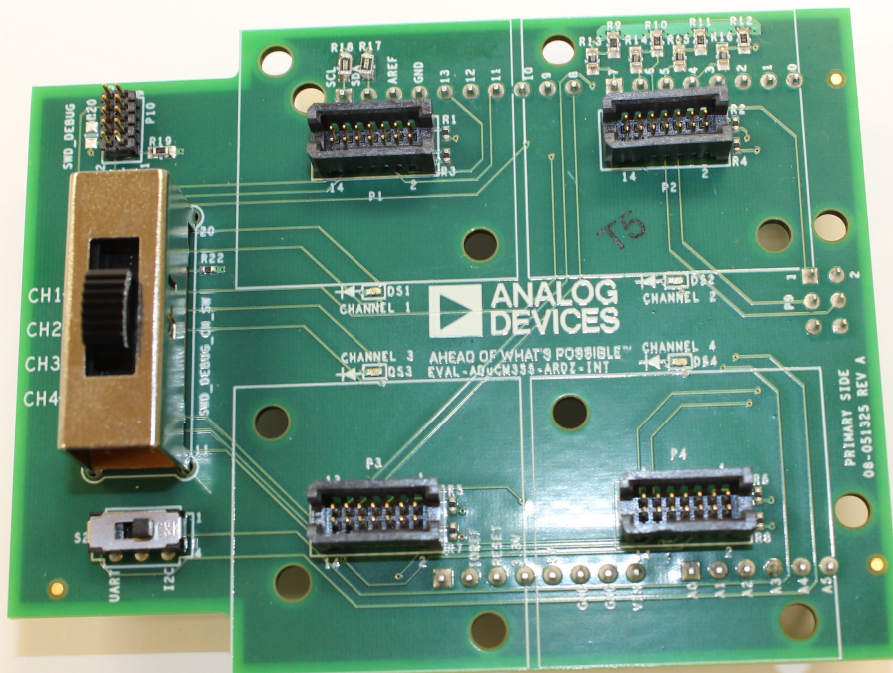


Boards Used

Following are the boards used:

Table of Contents

- ♦ [EVAL-CN0429-EBZ Overview](#)
- ♦ [Boards Used](#)
- ♦ [Features](#)
- ♦ [Documents Needed](#)
- ♦ [Equipment Required](#)
- ♦ [Test Setup Functional Block Diagram](#)
- ♦ [Hardware Setup](#)
 - ♦ [Connecting the Electrochemical Gas Sensor](#)
 - ♦ [Temperature & Humidity Sensor](#)
 - ♦ [Connector P1](#)
 - ♦ [Switch Configurations](#)
- ♦ [Software Setup](#)
 - ♦ [Setting up the Serial Terminal Software to Communicate with the Boards](#)
- ♦ [Example of System Setup](#)
- ♦ [Programming the CN0429](#)
- ♦ [IDE Setup](#)
- ♦ [Command Line Interface \(CLI\)](#)
 - ♦ [Getting Started](#)
 - ♦ [CLI Command Set](#)
 - ♦ [Select the Sensor](#)
 - ♦ [Sensor Configuration](#)
 - ♦ [Reading Sensor Data](#)
 - ♦ [Sensor Diagnostics](#)
- ♦ [Schematic, PCB Layout, Bill of Materials](#)
- ♦ [Software](#)



EVAL-M355-ARDZ-INT Arduino Shield Board



EVAL-ADICUP3029 Board



EVAL-CN0429-EBZ Gas Sensor Daughter Board

Features

- Capable of measuring any electrochemical gas sensor in a suitable package
- Up to 4 sensor boards can be connected for measurements simultaneously
- Gas sensor daughter boards includes temperature & humidity sensor
- Electrochemical Impedance Spectroscopy and Bias Voltage Pulse Test capabilities

Documents Needed

- [ADuCM355 Data Sheet](#)
- [CN0429 Circuit Note](#)
- [EVAL-ADICUP3029 User Guide](#)

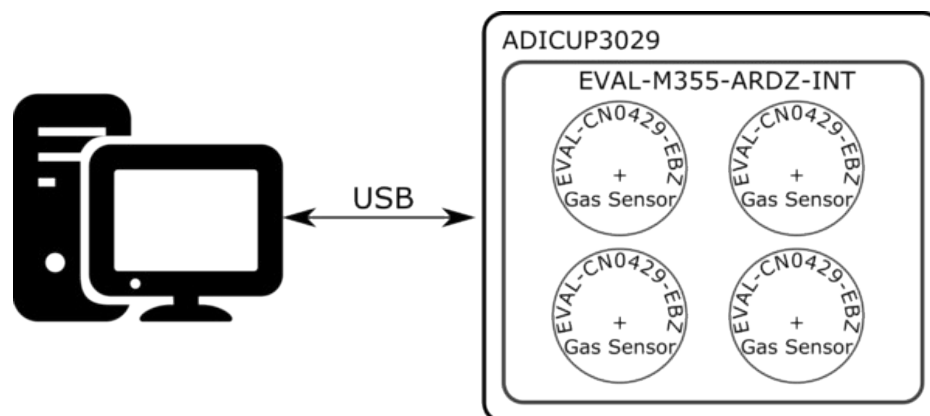
Equipment Required

- **Hardware**
 - EVAL-CN0429-EBZ Gas Sensor Daughter Board
 - EVAL-ADICUP3029 Base Board
 - EVAL-M355-ARDZ-INT Interposer Board
 - Micro USB Cable
 - Electrochemical Gas Sensor in a suitable form factor
- **Software**
 - PC with a [USB](#) port and Windows 7 (32-bit) or higher
 - Serial Terminal Software (Putty/TeraTerm or similar)
 - 3-Terminal or 4-Terminal Electrochemical Gas Sensor

Test Setup Functional Block Diagram

To set up the circuit for evaluation, consider the following steps:

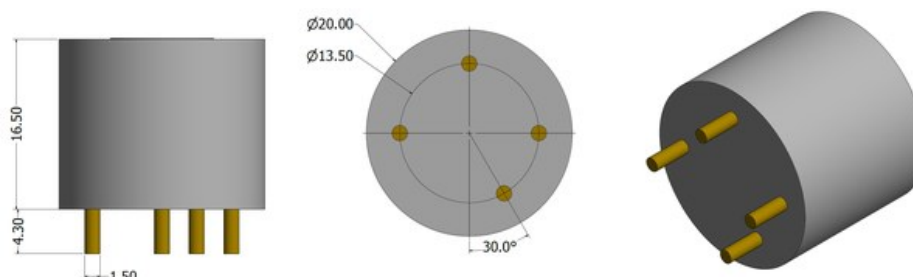
1. The EVAL-CN0429-EBZ ([▶ CN0429](#) Shield board) connects to [▶ EVAL-M355-ARDZ-INT](#) interposer board
2. That combination setup above, plugs directly into the [▶ EVAL-ADICUP3029](#) base board
3. Connect the system to the PC using [USB](#) cable
4. Connect the sensors to the dedicated connectors on the EVAL-CN0429-EBZ



Hardware Setup

Connecting the Electrochemical Gas Sensor

The CN0429 is compatible with electrochemical gas sensors in “4-series” form factor. 2-, 3- and 4-electrode sensors are supported. See the drawing below for form factor dimensions.



Examples of sensors supported by this reference design:

- CiTiceI® 4CF+
- Alphasense CO-A4
- DD Scientific GS+4CO

Temperature & Humidity Sensor

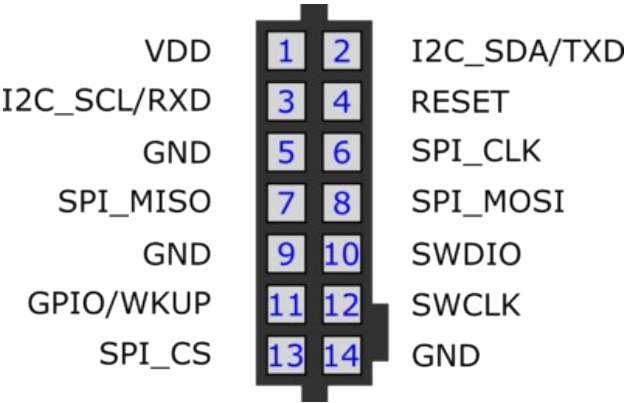
The CN0429 gas sensor daughter board includes [IDT HS3002](#) temperature & humidity sensor. This sensor is connected to a virtual I2C bus of the ADuCM355 microcontroller. Temperature and humidity values could be used to implement compensation techniques for an electrochemical gas sensors. For more information about how to obtain sensor data, please refer to the Software section.



add link to Software section here

Connector P1

Connector P1 is used for interfacing with the CN0429 board. It is used to provide power to the board and also to communicate with it using I2C, SPI or UART interfaces. There is also one GPIO available, which can be used as an interrupt input to wake-up the ADuCM355 or provide any other interrupt function.



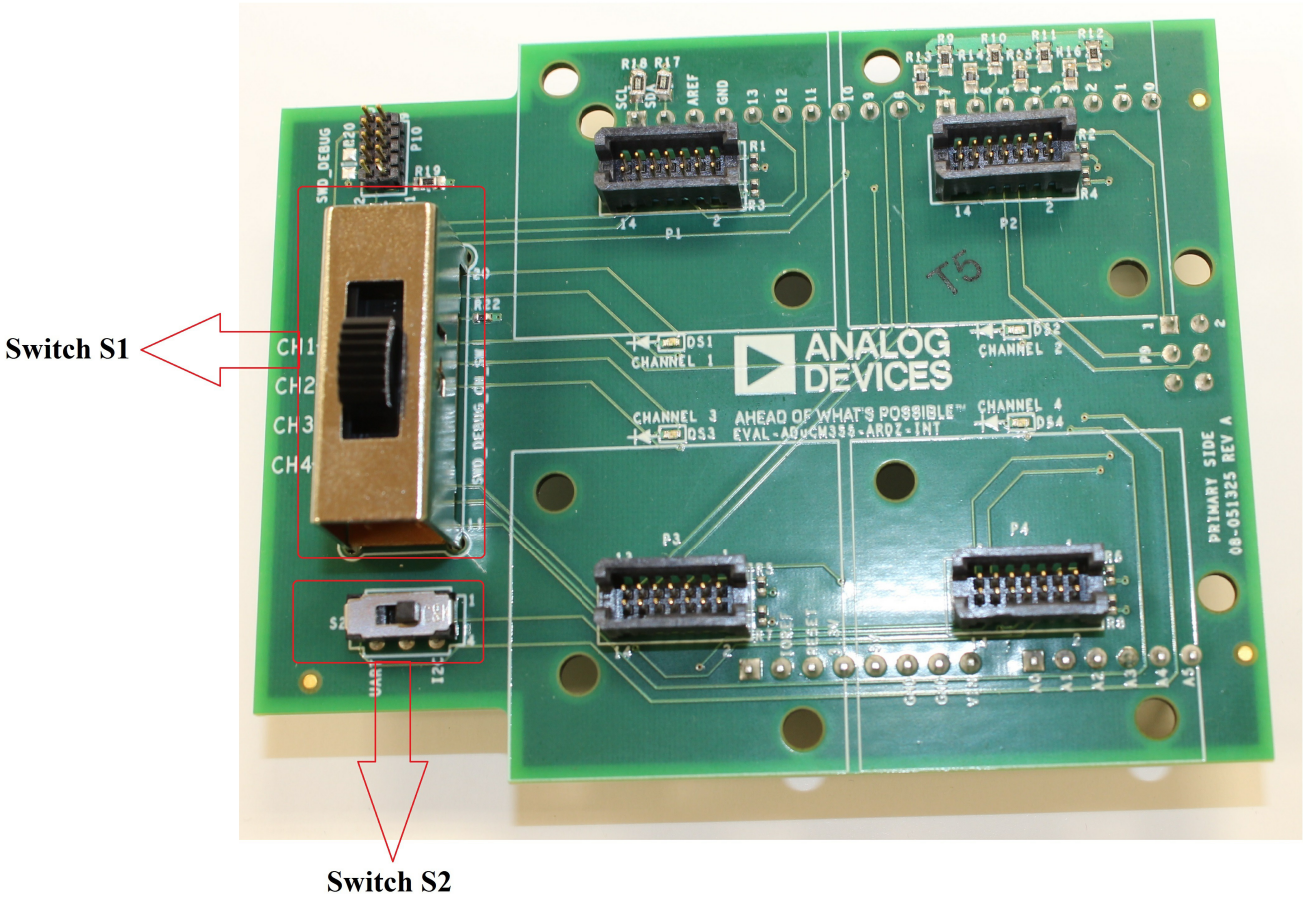
P1 connector pinout

The connector used on the CN0429 board is [Samtec SFM-107-02-L-D](#). The mating connector for it is either TFM-107-02-L-D (surface mount) or TFM-107-01-L-D (through hole).

Switch Configurations

Switch Configuration for EVAL-M355-ARDZ-INT Arduino Shield Board

There are two switches present on the [EVAL-M355-ARDZ-INT](#) board, namely switches S1 and S2.



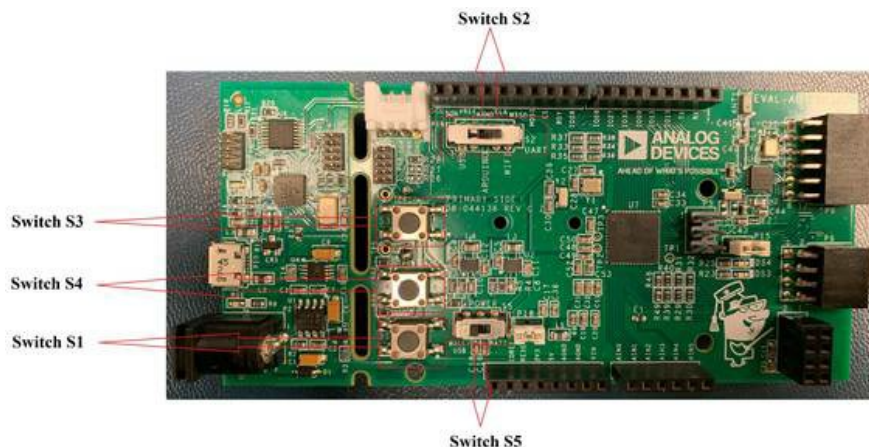
- Switch S1 selects which channel on the EVAL-M355-ARDZ-INT that is currently programmable. This is only needed/used when flashing

custom firmware onto the EVAL-CN0429-EBZ

- Switch S2 selects the communication method when SPI(default communication method) is not being used. So UART and I2C are possible communication modes back to the processor, as long as the firmware on the EVAL-CN0429-EBZ is set accordingly.

Switch Configurations for EVAL-ADICUP3029 Board



There are 5 switches present on the  EVAL-ADICUP3029 board, namely switches S1, S2, S3, S4 and S5.

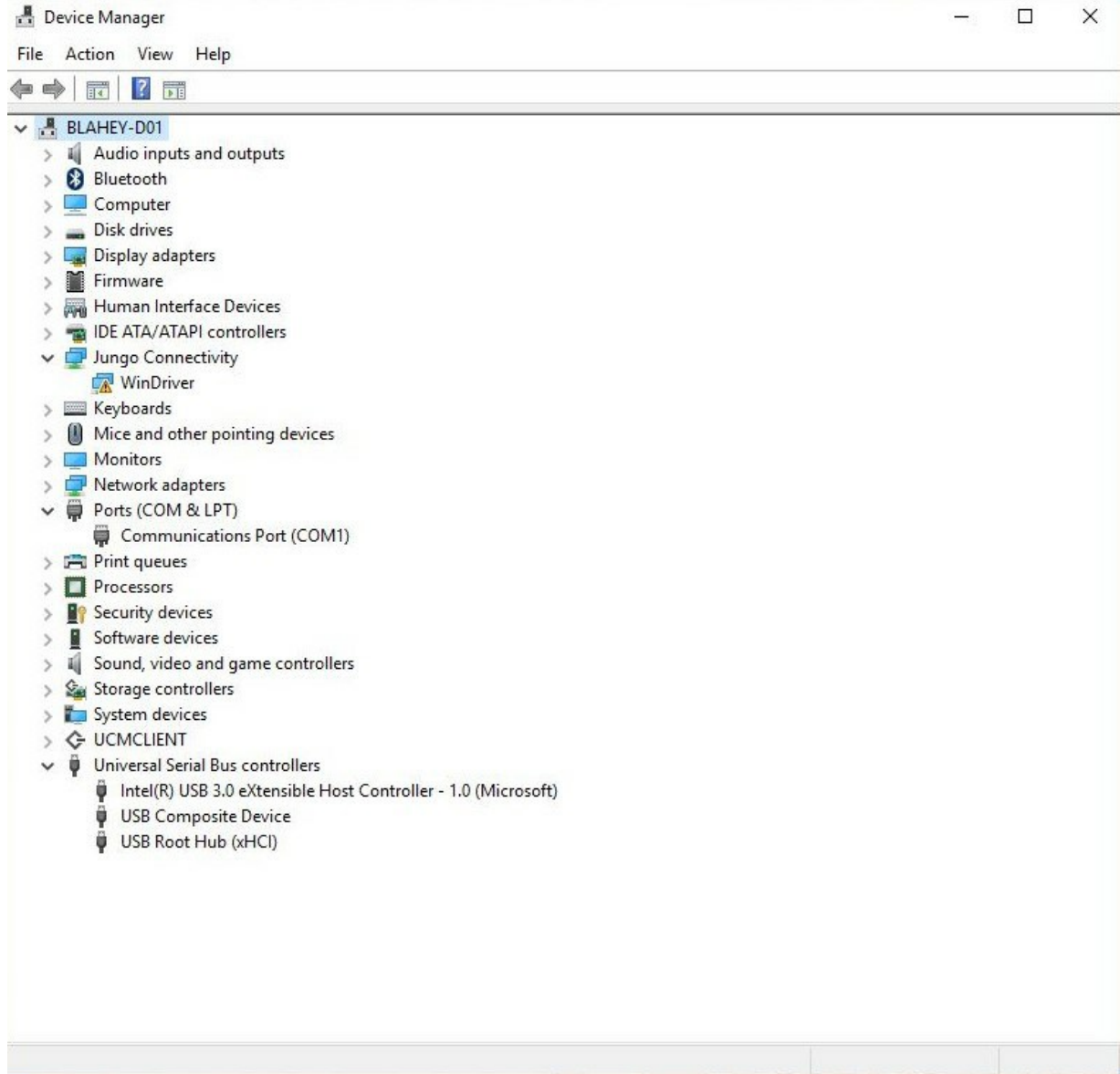


- Switch S2 routes the UART signals (Tx/Rx) from the ADuCM3029 and has three(3) options, 'USB', 'Arduino' and 'WiFi'. While in USB mode, the board communicates UART over the USB with the PC. This is the recommended setting for water quality measurements. When the switch is on Arduino mode, it establishes a UART communication with the Arduino connects on the ADICUP3029. WiFi mode allows the user to communicate with the plugin WiFi module available with the ADICUP3029.
- Switch S5 selects the power source and has two(2) options, 'Wall/ USB' and 'Battery'. These options allows the user to choose between powering the board using USB/DC wall supply (choosing Wall/ USB on switch S5) or using 2 AAA Batteries (choosing Batt on switch S5).
- Switch S1 is a reset switch, used to reset the board.
- Switch S3 is a boot switch, used to reboot the board by pressing the reset and boot switch together.
- Switch S4 is a WiFi reset switch, used to reset the externally connected WiFi board.

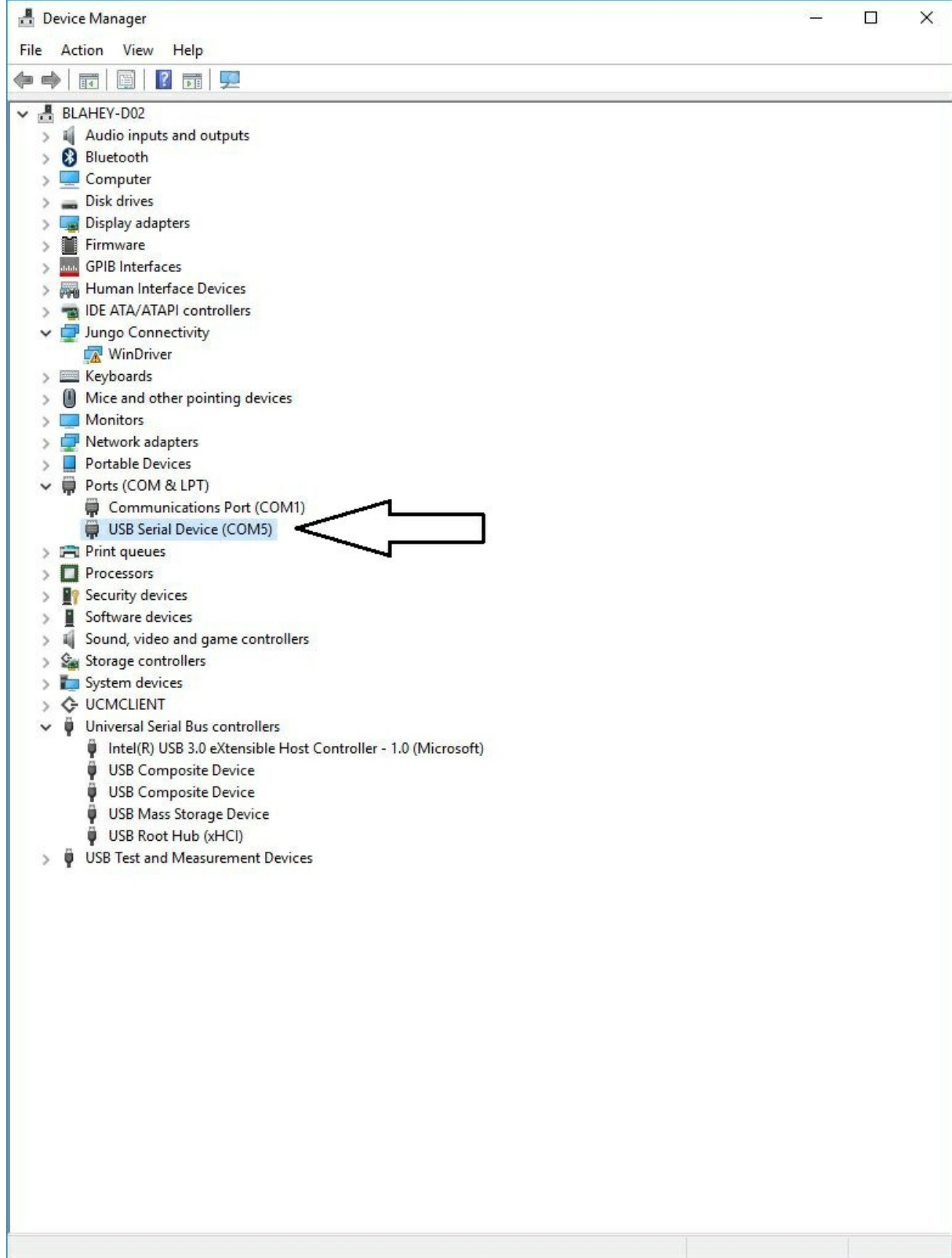
Software Setup

Setting up the Serial Terminal Software to Communicate with the Boards

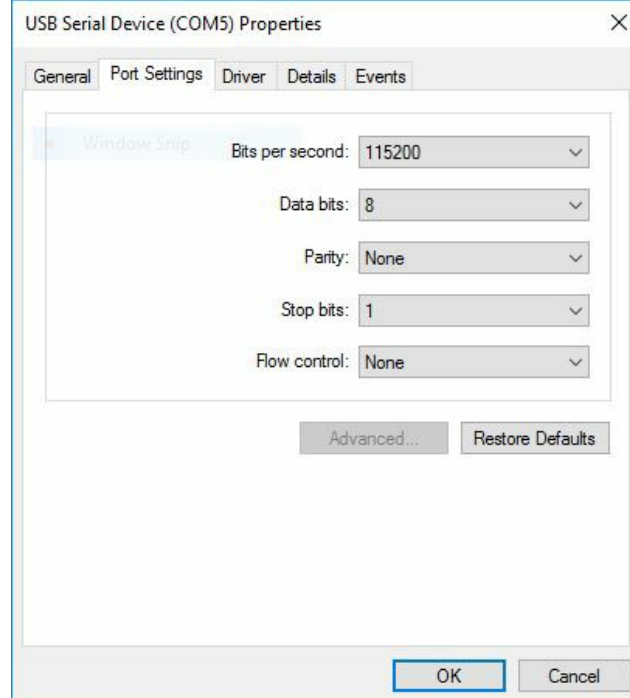
The  EVAL-CN0429-EBZ board and  EVAL-ADICUP3029 board come with a pre-installed base code. The user needs to install a serial terminal software on their PC to communicate with the board. The recommended software is PuTTY which is easily available over the internet for free installation. Once PuTTY is installed, user needs to go to the Device Manager terminal through the Start option in Windows. The Device Manager terminal opens as follows:



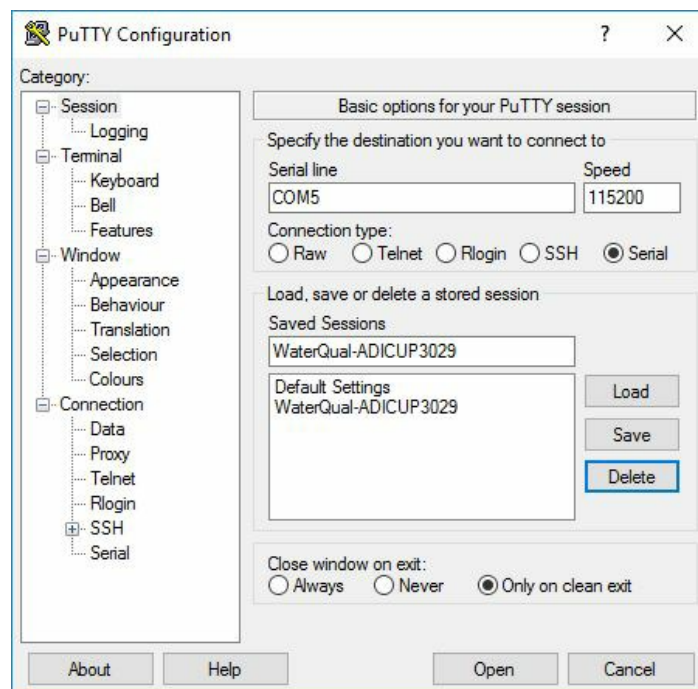
The user is then advised to plug in the [EVAL-ADICUP3029](#) board into the PC using the USB cable that goes into the USB port. As soon as the board is detected by the PC, the Device Manager displays the board and the port it is connected to.



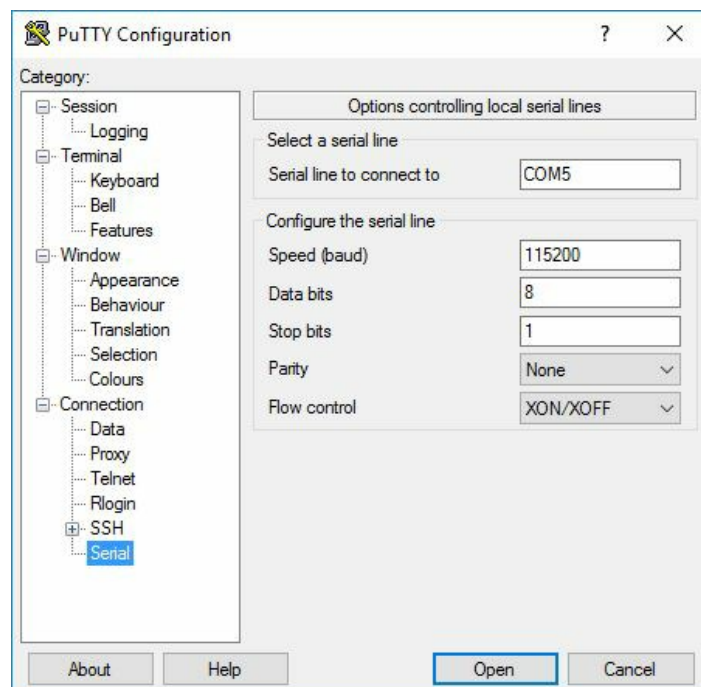
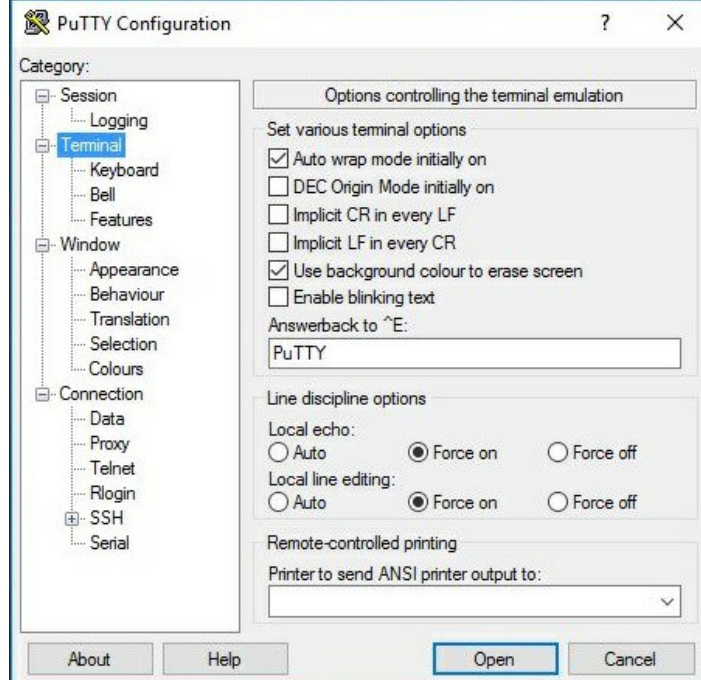
The new options that appear in the Device Manager window, as shown in the picture above, shows that the PC has detected the [EVAL-ADICUP3029](#) board. The user is then advised to note the port number of the USB Serial Device (COM5 as seen above). With a right click on the USB Serial Device (COM5), the user can then access the properties of the board connected. Note the speed shown here is 115200.



Once these steps have been followed, the user can now to start the PuTTY application and enter certain values in the configuration window. The three important settings that needs to be defined in the configuration window are the Session settings, the Terminal settings and the Serial settings. In the Session window, the speed should be set at 115200 and the serial line is the port number taken from Device manager (COM5 in our case).



The user can save these settings by a particular name (we call it WaterQual_ADICUP3029) and load them whenever needed. Other important configurations include the terminal settings and the connection serial settings. They should be set as follows before launching the Putty application:



Once all the configurations have been defined, launch the PuTTY terminal by selecting the 'Open' option. After the PuTTY terminal launches, write 'help' and press Enter to see the different commands in the PuTTY terminal for various gas measurements. See the [Command Line Interface](#) section for more details.

Example of System Setup

In this section, step by step instructions are provided to configure the system and the sensor for correct operation. It is recommended to always use this procedure.

Preparing the system

1. Plug the [EVAL-M355-ARDZ-INT](#) board into the [EVAL-ADICUP3029](#) board.
2. Plug the EC sensor to the [EVAL-CN0429-EBZ](#) board.
3. Plug the EVAL-CN0429-EBZ board with EC sensor to any of the four positions on the EVAL-M355-ARDZ-INT board.
4. Connect the assembled system to the PC with MicroUSB cable.
5. Open PUTTY serial terminal, set the COM port to 115200 8N1 configuration, and enable forced local echo.
6. Connect to the board using PUTTY terminal.
7. Press the RESET switch on the EVAL-ADICUP3029 board. It is under the EVAL-M355-ARDZ-INT board, but reachable.
8. Observe initialization messages of the system in PUTTY terminal window.
9. Once initialized, you can start using the system.

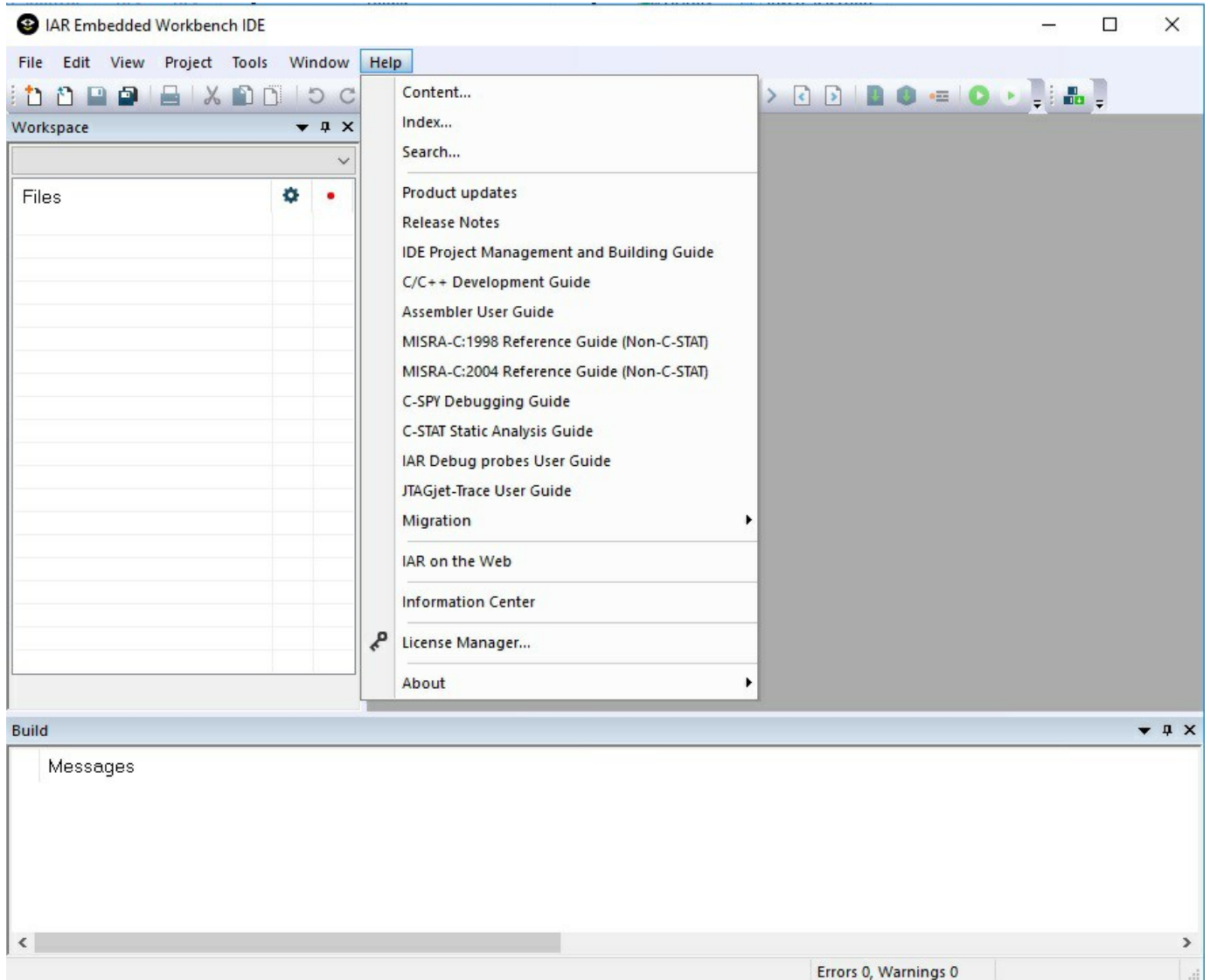
The EVAL-CN0429-EBZ boards come pre-programmed and pre-configured. Sensor configuration needs to be modified according to the datasheet of the sensor used. Refer to the [Sensor Configuration](#) section for more details.

Programming the CN0429

The [Eval-CN0429-EBZ](#) board comes with a pre-loaded base code. However if the user wishes to modify the firmware for custom measurements, they need to download and install IAR ARM 8.30.2 (or above) workbench. Along with this the user also needs to download the ADuCM355 support package, cut traces for the Serial Wire Debug (SWD) line and use the included cable to connect the debugger to the shield board.

IDE Setup

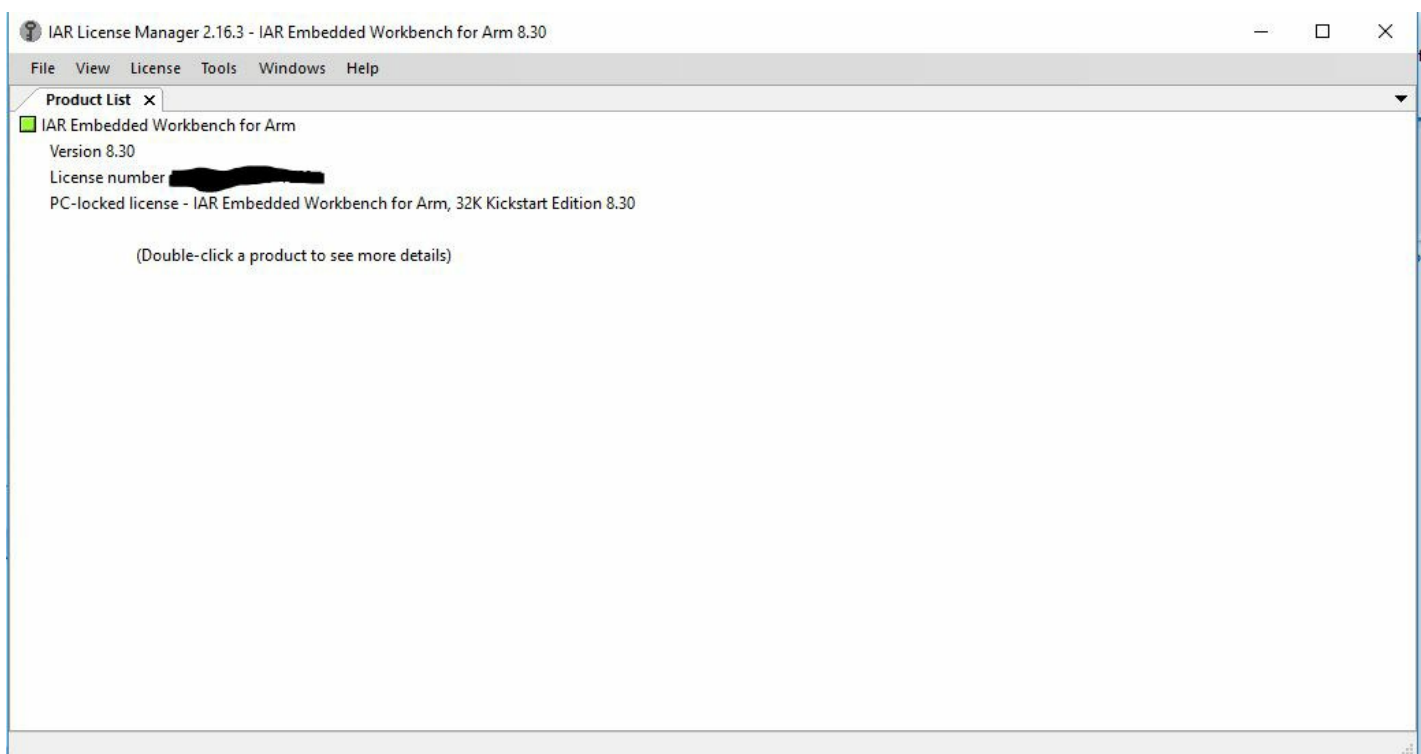
1. Install IAR Embedded Workbench for ARM
 - Please visit <https://www.iar.com/> to download IAR Embedded Workbench for ARM (version 8.30.2 or above)
2. License Installation
 - Once downloaded, the IAR window pops up as shown below. To activate the IAR workbench, click on the help tab.



- Then select the License Manager option. This will open a tab asking the user to register for a license key.



- When the user clicks on register, they are redirected to IAR website wherein they need to fill out their details to get access to a license number. Upon filling up the details, the user will receive a license number which has to be entered back in the tab above then hit Next. This completes the IAR setup and the user can validate their IAR license by looking at this window that pops up in the License Manager option now.

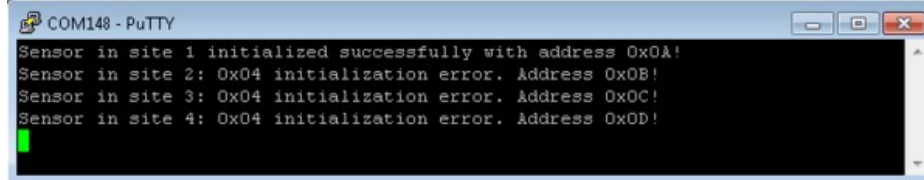


- The user now needs to load the ADuCM355 support package and modify the code by reprogramming it. After cutting the traces and programming the boards, the cable needs to be connected to the [EVAL-ADICUP3029](#) board again to allow USB communication.

Command Line Interface (CLI)

Getting Started

The CN0429 reference design is controlled using a command line interface. Any serial port terminal application can be used, e.g. Putty, TeraTerm. Configure the COM port to 115200/8N1 (115200 baud, 8 data bits, no parity bit and 1 stop bit). Once the system is programmed and connected to the PC, connect to it using the serial terminal application of your preference. The system is automatically initialized upon power-up and will print initialization report (example below).



Note: The gas sensor daughter boards require ~10 seconds to calibrate and start up properly. It is recommended that ~10 seconds after power is applied to the system, the RESET button S1 is pressed on the ADICUP3029 board to ensure proper initialization of the system.

CLI Command Set

The following table lists all available commands for the CN0429. Some of the commands are global – meaning they are applicable to all detected sensors at once. Others require user to first select a sensor and the command will then be used for the selected sensor only.

Syntax	Parameter	Description	Usage
help		Print command set help.	GLOBAL
defaultsensor	= 1 – 4	Select sensor site used by local commands.	
sensorsconnected		Print list of detected sensors.	
readconfigs		Read configuration of all sensors.	
readsensors		Read sensor data of all sensors.	
setupdaterate	= 1 – 3600	Set update rate for “readsensors” in seconds. Default = 1.	
stopread		Stop sensor data reading.	
readtemp		Read temperature from the on-board temperature sensor.	LOCAL Sensor site needs to be selected prior to using these commands
readhum		Read humidity from the on-board humidity sensor.	
setmeas	= 50 - 32000	Set ADC sampling time in milliseconds. The ADC performs average of 10 samples at configured interval Default = 500.	
startmeas		Start ADC sampling at configured interval.	
stopmeas		Stop ADC sampling the sensor.	
setrtia	= 0 - 26	Select internal RTIA resistor to set <u>TIA</u> gain.	
setrload	= 0 - 7	Select internal RLOAD resistor for the sensor.	
setvbias	= -1100 – 1100	Set bias voltage of the sensor in <u>mV</u> .	
setsensitivity	= nA/ppm	Set sensor sensitivity in nA/ppm.	
runeis		Run Electrochemical Impedance Spectroscopy test (EIS).	
readeis		Read results of EIS test.	
readeisfull		Read full set of results of EIS test, including all impedances and magnitudes.	
readrcal		Read value of internal 200R calibration resistor.	
runpulse		Run the pulse test. Amplitude and Duration need to be set first!	
readpulse		Read results of the pulse test.	
pulseamplitude	= 1 – 3	Pulse test amplitude in <u>mV</u> (typically 1 <u>mV</u>).	
pulseduration	= 1 - 199	Pulse test duration in milliseconds. Must be less than 200 msec.	

Select the Sensor

To be able to use any local command, a sensor needs to be selected first. Use following command to list all available (detected) sensors.

```
sensorsconnected
```

Afterwards, select the desired sensor, for example using:

```
defaultsensor 1
```


This command selects the sensor site. When using local commands, these apply to the selected site only. When no sensor site is selected, or there is no sensor connected at the selected site, an ERROR message will be displayed when trying to execute a local command.

Sensor Configuration

The ADuCM355 firmware comes with pre-programmed default gas sensor configuration. In order for the sensor to work correctly, this configuration needs to be changed according to the sensor specification which can be found in the sensor datasheet. Following configuration parameters need to be set using the command line interface:

- Bias voltage [mV]
- Sensor sensitivity [nA/ppm]
- RTIA [ohm]
- RLOAD [ohm]
- Measurement time [msec]

Note: Select the correct sensor site prior to using any of the following commands.

Bias voltage

Bias voltage is the voltage applied between RE and WE (SE) electrode of the gas sensor. Its value can be found in the sensor datasheet.

Example: setting bias voltage to 20 mV

```
setvbias 20
```

Sensor sensitivity

Sensor sensitivity value is used to calculate the gas concentration from the measured current produced by the sensor. If the sensor datasheet only states a range of currents instead of an exact value, please contact the sensor manufacturer and ask for more accurate value.

Example: setting setsensitivity to 73 nA/ppm

```
setsensitivity 73
```

TIA Gain - RTIA

The RTIA resistor value should be selected so that it maximizes the ADC input range of ±900 mV. The RTIA value is calculated using following equation:

$$R_{TIA} = \frac{0.9\text{ V}}{\text{Sensitivity} \times \text{Max_Range}}$$

where:
0.9 V is ADC input range,
Sensitivity is defined as nA/ppm and
Max_Range is the sensor's maximum range in ppm.

From the table below, select the closest lower value to the calculated RTIA value.

value	0	1	2	3	4	5	6	7	8
Resistor [ohm]	0	200	1k	2k	3k	4k	6k	8k	10k
value	9	10	11	12	13	14	15	16	17
Resistor [ohm]	12k	16k	20k	24k	30k	32k	40k	48k	64k
value	18	19	20	21	22	23	24	25	26
Resistor [ohm]	85k	96k	100k	120k	128k	160k	196k	256k	512k

Example - CO sensor:
Sensitivity = 73 nA/ppm
Max_Range = 500 ppm

$$R_{TIA} = \frac{0.9}{73 \times 10^{-9} \times 500} = \frac{0.9}{3.65 \times 10^{-5}} \approx 24.7\text{ k}\Omega$$

Closest lower value to 24.7 kΩ is 24 kΩ, i.e. = 12 .

```
setrtia 12
```

Load Resistor - RLOAD

The load resistor value can be obtained from the sensor datasheet. If there is no exact match between the datasheet value and table below, select the closest value.

value	0	1	2	3	4	5	6	7
Resistor [ohm]	0	10	30	50	100	1k6	3k1	3k6

Example - CO sensor:

Datasheet value is 47 Ω . The closest available value is 50 Ω :

```
setrload 3
```

Measurement Time

The EC sensor is sampled by the ADC with a certain time period, at which it performs an average of 10 samples, each 2.2 ms long. Default value of this time period is 500 ms, but can be configured to in the range of 50 - 32 000 ms.

Example: setting measurement period to 1000 ms

```
setmeastime 1000
```

Reading Sensor Configuration

Once the configuration has been performed, use the *readconfigs* command to verify that configuration is complete. This command reads configuration of all detected sensors.

```
readconfigs
```

Reading Sensor Data

Once all the sensors are configured, sensor data can be obtained. Temperature and humidity data from on-board sensor can be read one value at a time with dedicated commands. Gas data are read as a continuous stream with configurable update rate.

Start ADC sampling

This local command instructs the ADC of selected sensor to start sampling the sensor at the time interval configured by the *setmeastime* command.

```
startmeas
```

Note: It is not needed to run this command as a part of sensor initialization. ADC sampling starts automatically, default value of 500 ms is used.

Stop ADC sampling

This local command instructs the ADC of selected sensor to stop sampling the sensor.

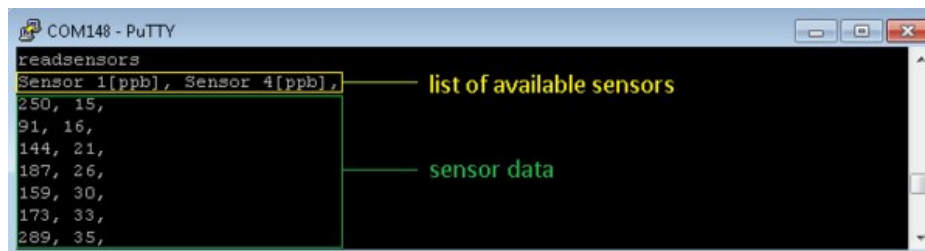
```
stopmeas
```

Start reading gas sensor data

Use this global command to start reading gas sensor data from all sensors.

```
readsensors
```

The data is read back as a continuous stream in a following format:



```
COM148 - PuTTY
readsensors
Sensor 1[ppb], Sensor 4[ppb],
250, 15,
91, 16,
144, 21,
187, 26,
159, 30,
173, 33,
289, 35,
```

Stop reading gas sensor data

Data stream can be stopped at any time using the *stopread* command.

```
stopread
```

Set stream update rate

The update rate of the data stream is configurable in the range of 1 – 3600 seconds. Stop the data stream before using this command.

Example: setting update rate to 5 seconds

```
setupdatarate 5
```

Read temperature

This local command prints temperature value from the temperature & humidity sensor on gas sensor daughter board.

```
readtemp
```

Read humidity

This local command prints humidity value from the temperature & humidity sensor on gas sensor daughter board.

```
readhum
```

Sensor Diagnostics

This reference design supports two diagnostics measurements of electrochemical sensors:

- Electrochemical Impedance Spectroscopy (EIS)
- Bias voltage pulse test

Electrochemical Impedance Spectroscopy

The EIS test is a diagnostics method where sensor impedance is measured at various frequencies of AC signal injected into the electrochemical sensor itself. The default set of frequencies supported by the sensor daughter board firmware is:

1kHz, 5kHz, 10kHz, 20kHz, 30kHz, 40kHz, 50kHz, 60kHz, 70kHz, 90kHz, 160kHz and 200kHz.

Note: It is possible to add additional frequencies to the EIS test. However, this has to be accounted for in both gas sensor daughter board firmware and ADICUP3029 firmware.

The test is executed by following these steps:

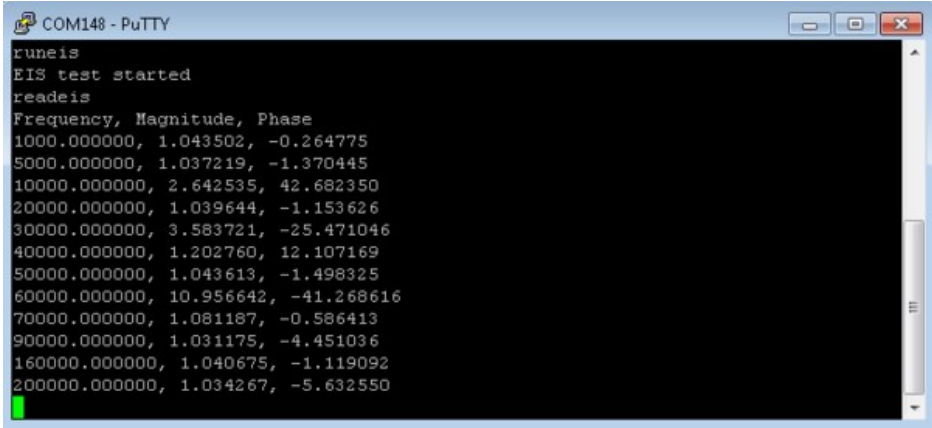
1. Run the EIS test (*runeis*)
2. Wait for at least 5 seconds until the test is finished
3. Read the test results (*readeis*)

```
runeis
```

```
readeis
```

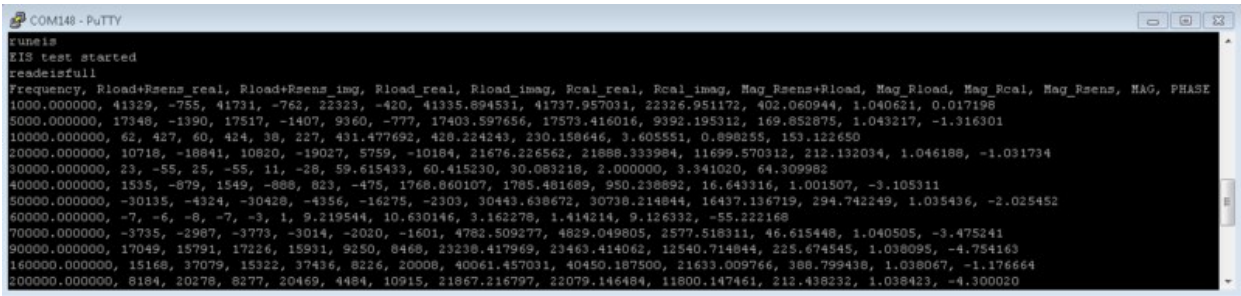
The results of the EIS test are in a format shown below. The values are comma delimited to enable importing the data to Microsoft Excel or similar tool for further processing.

Bode magnitude and Bode phase values are printed for every measured frequency.



To obtain **full set** of results including all magnitudes and impedances, use the following command:

```
readeisfull
```



Bias voltage pulse test

This diagnostics method can be used, for example, to detect open circuit, i.e. unplugged sensor. This proved to be a common issue with some portable instruments which can fall on the ground and sensor can disconnect from its socket.

The principle of this test is applying small voltage pulse (typically 1 mV for <200 ms) to the sensor electrodes and measuring the current response to this pulse.

Test execution procedure follows these steps:

- 1. Set pulse amplitude (*pulseamplitude*)
- 2. Set pulse duration (*pulseduration*)
- 3. Run the pulse test (*runpulse*)
- 4. Wait for at least 5 seconds
- 5. Read the test results (*readpulse*)

Example: setting pulse amplitude to 1 mV

```
pulseamplitude 1
```

Example: setting pulse duration to 100 ms

```
pulseduration 100
```

```
runpulse
```

```
readpulse
```

The example below shows portion of the pulse test results. The values are comma delimited to enable importing the data to Microsoft Excel or similar tool for further processing.


```
COM148 - PuTTY
runpulse
Pulse test started
readpulse
time [msec], current [uA]
0, 42.70
110, 42.70
220, 43.40
330, 43.40
440, 43.40
550, 44.10
660, 44.10
770, 44.80
880, 45.50
990, 44.80
1100, 44.80
1210, 45.50
1320, 45.50
1430, 46.20
1540, 46.90
1650, 46.20
1760, 46.90
1870, 47.60
1980, 47.60
2090, 47.60
2200, 48.30
2310, 48.30
2420, 48.30
2530, 49.00
2640, 49.00
2750, 49.00
2860, 49.70
2970, 49.70
```

Schematic, PCB Layout, Bill of Materials

EVAL-CN0429-EBZ Design & Integration Files



- Schematics
- PCB Layout
- Bill of Materials
- Allegro Project

Software

-  [CN0429 ADuCM355 Source Code](#)
-  [CN0429 ADICUP3029 Source Code](#)

End of Document

[resources/eval/user-guides/circuits-from-the-lab/cn0429.txt](#) · Last modified: 06 Dec 2018 12:44 by MRaninec

15,000

Problem Solvers

4,700+

Patents Worldwide

125,000

Customers

50+

Years

Ahead of What's Possible

Analog Devices is a global leader in the design and manufacturing of analog, mixed signal, and DSP integrated circuits to help solve the toughest engineering challenges.

Analog Devices. Dedicated to solving the toughest engineering challenges.

[See the Innovations](#)

SOCIAL



QUICK LINKS

[About ADI](#)

[Partners](#)

LANGUAGES

[English](#)

NEWSLETTERS

[Interested in the latest news and articles about ADI](#)



[Analog Dialogue](#)
[Contact us](#)
[News Room](#)
[Sales & Distribution](#)

[Careers](#)
[Investor Relations](#)
[Quality & Reliability](#)

[简体中文](#)
[日本語](#)
[Русский](#)

products, design tools, training and events?
Choose from one of our 12 newsletters that match
your product area of interest, delivered monthly or
quarterly to your inbox.

[Sign Up](#)