

## 1 Introduction

This application note provides an overview of how to use RT500 SDK software with EVK hardware to validate the RT500 built-in temperature sensor using the ADC controller. The ADC input channel 7 is mapped to an internal temperature sensor.

ADC watermark interrupt would be asserted once the number of data words stored in the ADC Result FIFO are greater than the watermark value. In ADC ISR, the watermark flag is cleared by reading the conversion result value. When the conversion done, two valid results are stored in the FIFO, and the temperature can be calculated using a specific formula within two results.

## 2 Temperature sensor overview

The temperature sensor transducer uses an intrinsic pn-junction diode reference and outputs a CTAT voltage (Complement To Absolute Temperature). The output voltage varies inversely with device temperature with an absolute accuracy of better than  $\pm 5$  °C over the full temperature range (-40 °C to +105 °C). The temperature sensor is only approximately linear with a slight curvature. The output voltage is measured over different ranges of temperatures and fit with linear-least-square lines.

### 2.1 Characteristics

The following table shows the temperature sensor static and dynamic characteristics.

**(VDDA\_BIAS = 3.3 V, All other supplies = 1.8 V)**

Symbol	Parameter	Conditions	Notes	Min	Typ	Max	Unit
DT <sub>sen</sub>	sensor temperature accuracy	T <sub>amb</sub> = -20 °C to 70 °C	1	-	-	2.77	°C
E <sub>L</sub>	linearity error	T <sub>amb</sub> = -20 °C to 70 °C		-	-	2.79	°C

1. Absolute temperature accuracy. Based on characterization. Not tested in production

### 2.2 Connection between ADC channel input and analog output

The following figure shows the connection between ADC channel input and analog output.

#### Contents

1	Introduction.....	1
2	Temperature sensor overview.....	1
3	Demo application.....	3
4	Conclusion.....	5
5	References.....	5
6	Revision history.....	5



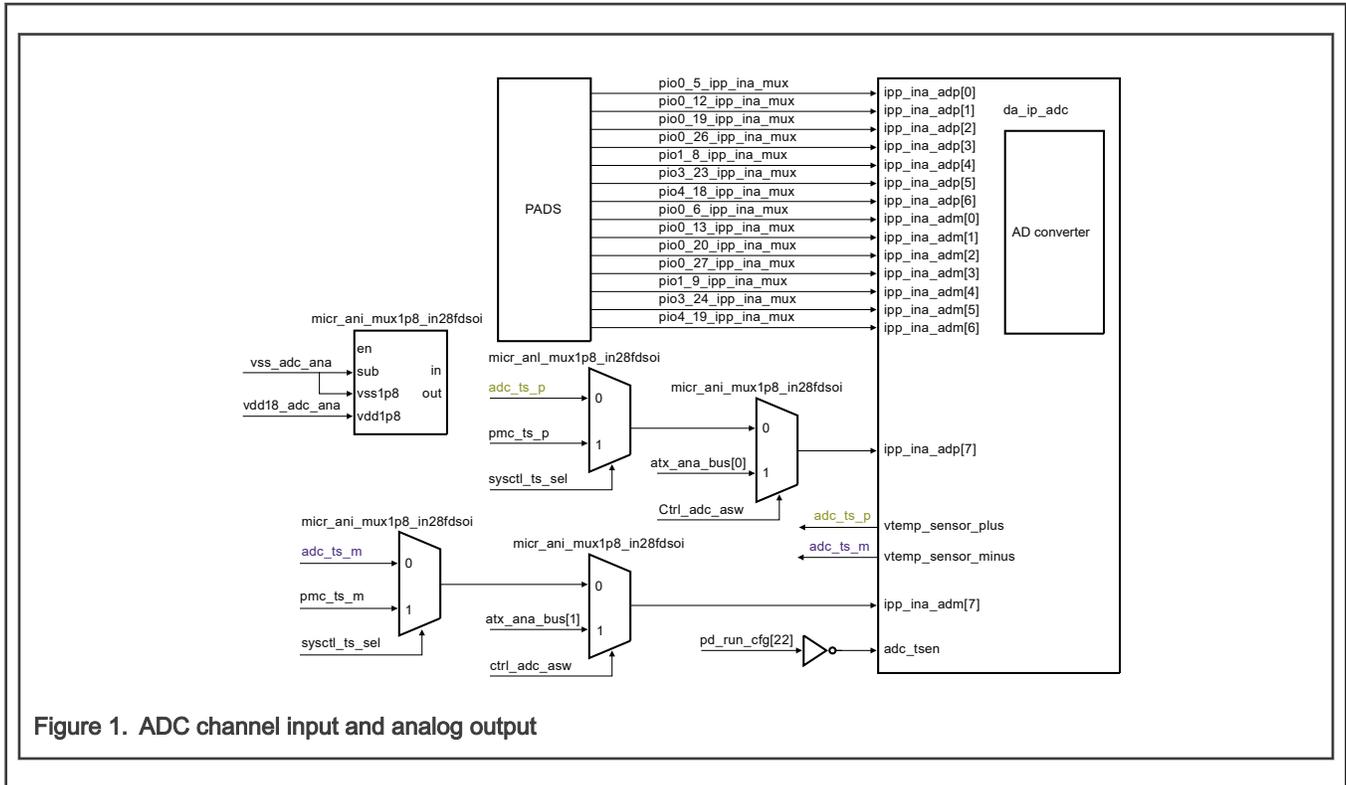


Figure 1. ADC channel input and analog output

## 2.3 Registers needed for ADC temperature sensor

### 2.3.1 Temperature sensor control (SYSCTL0\_TEMPSENSORCTL)

This register enables the on-chip temperature sensor to be measured by the ADC.

Field	Description
31-1	Reserved
—	
Field	Description
0	Temperature Sensor Source
TSSRC	0 - ADC Built-in Temperature Sensor 1 - PMC Temperature Sensor

### 2.3.2 Run configuration register 1 clear (SYSCTL0\_PDRUNCFG1\_CLR)

Writing a 1 to a bit position in this register clears the corresponding position in PDRUNCFG1. This is a write-only register.

22 ADC_LP	<p>ADC low power mode</p> <p>To minimize power consumption in deep_sleep or deep_powerdown mode both ADC_LP and ADC_PD bits must be set.</p> <p>0 - No effect</p> <p>1 - Clears the PDRUNCFG0 Bit</p>
21 ADC_PD	<p>ADC analog functions</p> <p>To minimize power consumption in deep_sleep or deep_powerdown mode both ADC_LP and ADC_PD bits must be set.</p> <p>0 - No effect</p> <p>1 - Clears the PDRUNCFG0 Bit</p>

## 2.4 Using typical parameter provided by the data sheet

Use the typical parameters provided by the RT500 data sheet to perform a temperature reading. An approximate transfer function describes the temperature sensor.

$$\text{Temp} = 25 - ((V_{\text{temp}} - V_{\text{temp}25})/m)$$

Where:

- $V_{\text{temp}}$  is the voltage of the temperature sensor channel at the ambient temperature.
- $V_{\text{temp}25}$  is the voltage of the temperature sensor channel at 25°C.
- $m$  is the temperature sensor slope.

In application code, the user executes a conversion on the temperature sensor channel and converts the conversion result to a voltage (VTEMP) based on the voltage reference used. The ambient temperature is then calculated using the above equation. VTEMP25 and the temperature sensor slope values are specified constant values from the ADC Electrical information in the device data sheet.

When converting on the temperature sensor channel, the following command selections must be configured:

- Differential mode (CMDLn[DIFF] = 0x1)
- Maximum averaging (CMDHn[AVGS] = 0x7)
- Maximum sample time (CMDHn[STS] = 0x7)

## 3 Demo application

### 3.1 Environment

#### 3.1.1 Hardware environment

- Board
  - MIMXRT595EVK
- Debugger
  - Integrated CMSIS-DAP debugger on the board
- Miscellaneous
  - 1 Micro USB cable
  - PC

- Board Setup
  - No special settings are needed

### 3.1.2 Software environment

- Tool chain
  - IAR embedded workbench 8.50.1 or MCUXpresso IDE v11.3.0 or Keil 5.29
- Software package
  - SDK\_2.9.0\_EVK-MIMXRT595

## 3.2 Project overview

1. Follow the Getting Started with MCUXpresso SDK for MIMXRT500 (available inside SDK→docs) to go through the steps for opening lpadc\_temperature\_measurement project (SDK\boards\evkmimxrt685\driver\_examples\lpadc\temperature\_measurement).
2. Open the file lpadc\_temperature\_measurement.c (lpadc\_temperature\_measurement\source) and find the values for the temperature slope and intercept that helps to calculate the ambient temperature based on the formula that is given in [Using typical parameter provided by the data sheet](#). These are specified constant values from the ADC Electricals information in the device data sheet.

```
#define DEMO_LPADC_TEMPERATURE_SLOPE          \
    -1.5738f /* Temperature sensor slope with the unit as mV/Celsius. \
             Please refer to the Data Sheet for details. */
#define DEMO_LPADC_TEMPERATURE_INTERCEPT   \
    809.55f /* The voltage of the temperature sensor channel at 0 Celsius with the unit as mV. \
            Please refer to the Data Sheet for details. */
```

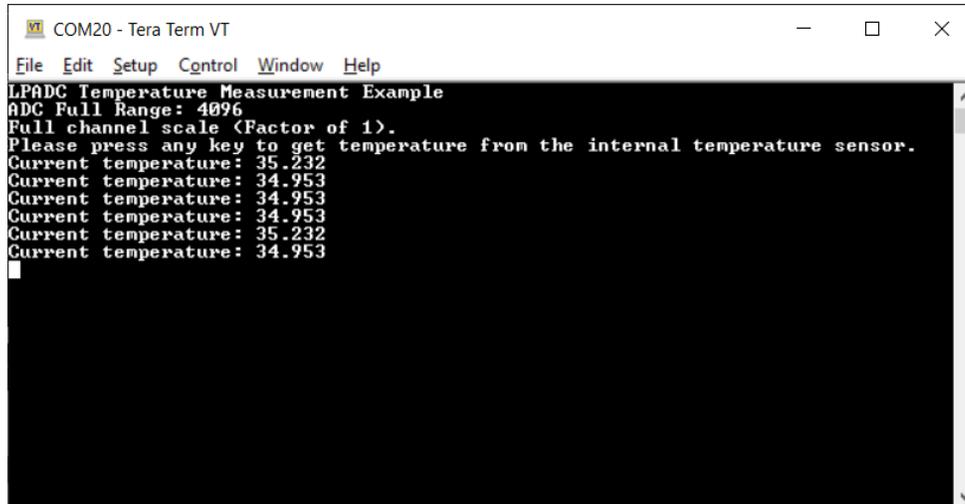
3. In SYSCTL>TEMPSENSORCTL, select between the two temperatures sensors available; ADC and PMC, the ADC is selected by default and no changes are required to use it. Note that the sensor is already enabled in the PDRUN configuration at the beginning of the main function.

```
/* Power up the ADC temperature sensor via SYSCTL. */
SYSCTL0->PDRUNCFG0_CLR = SYSCTL0_PDRUNCFG0_ADC_TEMPSNS_PD_MASK;
```

4. The STS bits of CMDH1 register are set to 7. So, the sample time is 131 (3 + 2STS) ADCK cycles since a sample time of over 35 is required. A long sample time allows higher impedance inputs to be accurately sampled. This is done by modifying the `sampleTimeMode` which is member of conversion commands configuration structure inside `LPADC_GetDefaultConvCommandConfig` function. This change is already done in the ADC configuration part of code.

```
g_LpadcCommandConfigStruct.sampleTimeMode = kLPADC_SampleTimeADCK131;
```

5. Follow the Getting Started with MCUXpresso SDK for MIMXRT500 (available inside SDK→docs) to go through the steps for building and running lpadc\_temperature\_measurement demo.
6. When running the demo, type any key in the debug console which triggers the conversion.
7. Result:



```

COM20 - Tera Term VT
File Edit Setup Control Window Help
LPADC Temperature Measurement Example
ADC Full Range: 4096
Full channel scale (Factor of 1).
Please press any key to get temperature from the internal temperature sensor.
Current temperature: 35.232
Current temperature: 34.953
Current temperature: 34.953
Current temperature: 34.953
Current temperature: 35.232
Current temperature: 34.953

```

## 4 Conclusion

The example shows how the SDK software with EVK hardware can be used to validate RT500 built-in temperature sensor using the ADC controller. It also shows, how the built-in temperature sensor can be used to indicate temperature value based on ADC output value.

## 5 References

1. [RT500 Reference Manual](#)
2. [RT500 Data Sheet](#)
3. MCUXpresso SDK Release Notes for EVK-MIMXRT595 (available inside SDK)
4. Getting Started with MCUXpresso SDK for EVK-MIMXRT595 (available inside SDK)

## 6 Revision history

Table 1. Revision history

Revision number	Date	Substantive changes
0	05/2021	Initial release

## How To Reach Us

### Home Page:

[nxp.com](http://nxp.com)

### Web Support:

[nxp.com/support](http://nxp.com/support)

**Limited warranty and liability**— Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: [nxp.com/SalesTermsandConditions](http://nxp.com/SalesTermsandConditions).

**Right to make changes** - NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Security**— Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at [PSIRT@nxp.com](mailto:PSIRT@nxp.com)) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, ICODE, JCOP, LIFE, VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro,  $\mu$ Vision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© NXP B.V. 2021.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: [salesaddresses@nxp.com](mailto:salesaddresses@nxp.com)

Date of release: 05/2021

Document identifier: AN13209