Release Notes for ADuCM302x Device Family Pack 4.0.0

1 Table of Contents

2 Introduction

Thank you for installing the ADuCM302x Device Family Pack (DFP). This document describes the changes for the ADuCM302x Device Family Pack 4.0.0. ADuCM302x Device Family Pack 4.0.0 is supported in Keil uVision, CrossCore Embedded Studio® (CCES) and IAR Embedded Workbench.

This product is subject to control under the US Export Regulation. The Export Control Classification Number (ECCN) is 5D002 (unrestricted).

3 Release Notes for ADuCM302x Device Family Pack 4.0.0

3.1 Differences between version 4.0.0 and prior versions

3.1.1 Flash Programmer

An application running on a board may disable Instruction SRAM. As a consequence, an attempt to download a new program in flash memory may fail because the ISRAM is disabled while this is where the flash programmer is executing. Consequently, SRAM_CTL reset is now enforced when flash programming with IAR.

3.1.2 Processor Files

Protect against risks of macro redefinitions in processor files.

3.1.3 Toolchain & CMSIS

- Pack built and tested with CCES 3.0.3, Keil MDK6 & IAR 9.70.1.
- GCC startup file updated for compatibility with ARM CMSIS Pack 6.3.0.

3.1.4 Debug & Tools

- Fixed OpenOCD compatibility issues.
- Updated debug logging to use `snprintf` (safer memory handling).
- Examples & Utilities
- Added LED blink support to the `hello_world` example
- Updated Pinmux tool with latest GPIO support.

3.1.5 Supported Hardware Updates

- Added support for:
    - EVAL-ADICUP3029
    - EV-COG-AD3029LZ

Note: ADuCM3029 EZ-KIT is not supported.

3.1.6 Device Support

- Added:
    - ADuCM3029-1
    - ADuCM3029-2

3.1.7 FreeRTOS

RTOS macros for critical section redefined to properly disable interrupts.

3.1.8 uC/OS-III

Support to disable SEM_DELETE using macro OS_CFG_SEM_DEL_EN added.

3.1.9 Flash Controller

Different flash controller macros, e.g. ADI_FEE_NUM_INSTANCES, FEE_FLASH_SIZE, FEE_BLOCK_SHIFT, FEE_MAX_NUM_PAGES, etc. now located in adi_flash.h

3.1.10 ADC

Code clearing bits in STATUS register in functions DmaFIFOManage and InterruptFIFOManage improved.

ADC API extended with function adi_adc_EnableIRQ to enable/disable interrupts.

3.1.11 Crypto

STAT register self-assignment instructions eliminated.

3.1.12 DMA

API extended with two functions:

- adi_dma_Enable: enable/disable the DMA
- adi_dma_ReInit: force adi_dma_init to fully re-execute

3.1.13 I2C

Rx Overflow and Tx Underflow detection fixed.

Functions adi_i2c_GetHWMaskedErrors and adi_i2c_SetHWMaskedErrors added to ignore HW errors, e.g. Rx Underflow, when desired.

Function adi_i2c_GetNumberOfDataProcessed added to get the number of bytes sent/received when a transaction is being serviced or an error occurred.

Support for bus clear operation added:

- Configuration parameters extended with ADI_I2C_CFG_MCTL_BUSCLR and ADI_I2C_CFG_MCTL_STOPBUSCLR to configure the I2C with a default behavior.
- Function adi_i2c_SetBusClear added to dynamically set/clear the BUSCLR and STOPBUSCLR bits.

Incomplete Rx Transmission detection added.

Fix the number of writable bytes in FIFO in functions commenceTransmit and commenceReceive.

Eliminate the risk of error caused by a semaphore posting in I2C interrupt handler.

3.1.14 PWR

Fix for DMA not being re-enabled when waking up from hibernation.

Function adi_pwr_EnableClockSource returns an error if a call to adi_gpio_InputEnable fails.

Function adi_pwr_ExitLowPowerMode clears the PWRMOD register along with bits SLEEPONEXIT and SLEEPDEEP in SCR register when exiting low power modes.

Resolved mapping error between ADuCM302x definitions and macros used in adi_pwr.c.

3.1.15 RTC

Function adi_rtc_GetISOENB added to check the ISENB bit value.

Fix for RTC interrupts cleared without being serviced.

Make sure all the pending interrupts are cleared when executing afi_rtc_Open.

3.1.16 SPI

SPI driver improved to eliminate the risk of Tx Underflow for DMA driven transactions

SPI DMA interrupt handling simplified and improved to handle Tx bytes number > Rx bytes number.

SPI configuration macros ADI_SPI_TRAP_RXOVR and ADI_SPI_TRAP_TXUNDR introduced to enable/disable RXOVR and TXUNDR error detection in SPI interrupt handlers. (Enable by default.)

3.1.17 UART

ACR register bits ABE, DNIEN and TOIEN are now cleared properly when calling adi_uart_EnableAutobaud to disable autobaud.

The UART DMA driven model has been extended with Ping-Pong and Scatter-Gather. This extension is disabled by default. Macro ADI_UART_DMA_EXTENSION_ENABLE must be set to 1 in the UART driver configuration file for this feature to be enabled.

Functions adi_uart_SetTxDmaMode and adi_uart_SetRxDmaMode to select PingPong or Scatter-Gather when ADI_UART_DMA_EXTENSION_ENABLE is enabled.

Protect concurrent access to IEN register: IEN is updated in UART interrupt handlers, which can cause issues when the register is being modified by UART functions called by users' applications.

API extended with function adi_uart_GetRX.

Macro guarded Rx Buffer extension to help users' callback functions to pad the Rx buffer when the number of bytes received is not a multiple of the number of bytes that triggers an interrupt.

Macro guarded Rx Buffer fast draining extension added.

Data transfer mode set to none when flushing Tx buffers.

4 Release Notes for ADuCM302x Device Family Pack 3.2.0

4.1 Differences between version 3.2.0 and prior versions

4.1.1 System Initialization

Disable ISRAM before calling SystemInit if ISRAM is to be disabled.

The adi_system_EnableRetention function now supports the combination of bank1 and bank2.

Replace obsolete bool_t type with bool for ADuCM3027. Correct invalid headers files in include statements for ADuCM3027.

Add missing pinmux file for ADuCM3027.

Support for relocating the vector table in SRAM re-introduced.

4.1.2 FreeRTOS

Default interrupt priority now configurable with non-0 values, a requirement when using FreeRTOS.

4.1.3 Crypto

Following the addition of the HMAC feature for the ADuCM4x50 family, some crypto examples stopped working properly. This fix was introduced in release 3.1.2.

The crypto driver has been made common to ADuCM302x and ADuCM4x50 families.

Allow 0-length input data for CCM mode.

4.1.4 PWR

Watchdog timer control register is now saved before entering hibernate and restored on wake up.

4.1.5 RTC

Function adi_rtc_SetAutoReloadValue now assign register SS1ARL as expected.

All the input capture channels should now be properly supported by the RTC driver.

Removed interferences between ADuCM4x50 specific support for RTC and ADuCM302x.

Eliminate compilation failures caused by modifications specific to ADuCM4x50.

4.1.6 UART

When a second buffer was submitted for a transmit, using another buffer, while a first buffer was already being transmitted (non-blocking), the first buffer was sent twice.

5 Release Notes for ADuCM302x Device Family Pack 3.1.2

5.1 Differences between version 3.1.2 and prior versions

5.1.1 Crypto

Following the introduction of struct member pHmacKey in struct ADI_CRYPTO_TRANSACTION, for ADuCM4050, member pKey has been renamed pAesKey.

5.1.2 GPIO

GPIO driver API extended with adi_gpio_GroupInterruptPolarityEnable to determine if the interrupts are generated on the rising or falling edge of the corresponding GPIO pin.

5.1.3 RTC Driver

RTC driver modified to eliminate the risk of counter overflows.

5.1.4 RTOS

The RTOS mapping has been extended with Micrium µC/OS-II.

5.1.5 UART Driver

UART driver updated for PIO Rx transfers to support all the FIFO trigger levels. (Previous versions supported 1-byte but not 4-byte/8-byte/14-byte.)

A minor change was required in adi_uart_SetRxFifoTriggerLevel for this modification: the hDevice parameter cannot be constant anymore as the Rx FIFO trigger level must be recorded.

ADuCM302x DFP 3.1.2

```
ADI_UART_RESULT adi_uart_SetRxFifoTriggerLevel(
                  ADI_UART_HANDLE const hDevice,
                  ADI_UART_TRIG_LEVEL const eTriglevel
                );
```

ADuCM302x DFP 3.1.0

```
ADI_UART_RESULT adi_uart_SetRxFifoTriggerLevel(
                  ADI_UART_CONST_HANDLE const hDevice,
                  ADI_UART_TRIG_LEVEL const eTriglevel
                );
```

6 Release Notes for ADuCM302x Device Family Pack 3.1.0

6.1 Differences between version 3.1.0 and prior versions

The main changes in version 3.1.0 is the extended support for IAR Embedded Workbench.

- ADuCM302x_DFP\3.1.0\ARM\config now including material to fully support ADuCM302x in CMSIS Pack, e.g. ICF files, DDF files, flash programmer.
- Source for building the flash programmer used by IAR available in ADuCM302x_DFP\3.1.0\ARM\src\flashloader\AnalogDevices\FlashADuCM3027 and ADuCM302x_DFP\3.1.0\ARM\src\flashloader\AnalogDevices\FlashADuCM3029.

6.2 Required Software

6.2.1 Keil uVision

To use this ADuCM302x Device Family Pack with Keil uVision, you must first obtain and install:

- Keil uVision MDK v6 or later with ARM Compiler version 6.3.0 or later.
- Segger J-Link LITE v5.10p or later.

Install the Keil software first, then install the Segger J-Link LITE software.

6.2.2 CrossCore Embedded Studio

To use this ADuCM302x Device Family Pack with CrossCore Embedded Studio, you must first obtain and install:

- CrossCore Embedded Studio 3.0.3 or later.

6.2.3 IAR Embedded Workbench

To use this ADuCM302x Device Family Pack with IAR Embedded Workbench, you must first obtain and install:

- IAR Embedded Workbench for ARM 9.70.1 or later.

6.3 Release Testing

6.3.1 Keil uVision

This ADuCM302x Device Family Pack has been tested with

ADICUP302x	Emulator
Eval-ADICUP3029 version 1.2	J-Link
EV-COG-AD3029LZ	CMSIS-DAP

6.3.2 CrossCore Embedded Studio

This ADuCM302x Device Family Pack has been tested with

ADuCM302x	Emulator
Eval-ADICUP3029 version 1.2	ICE-2000
EV-COG-AD3029LZ	J-Link

6.3.3 IAR Embedded Workbench

This ADuCM302x Device Family Pack has been tested with

ADICUP302x	Emulator
Eval-ADICUP3029 version 1.2	J-Link
EV-COG-AD3029LZ	CMSIS-DAP

6.4 License Checking

Use of ADuCM302x Device Family Pack software is subject to the Software License Agreement presented during installation.

The details of this Software License Agreement can be found in the CMSIS pack installation directory, in AnalogDevices\ADuCM302x_DFP\4.0.0\License.

6.5 Release Content

This release contains the following sets of components:

- Source files for the ADuCM302x device family drivers. These components are authored by Analog Devices, for use on the ADuCM302x processor.
- Toolchain support. These components are authored by Analog Devices, and are installed into the toolchain to configure it to recognize the ADuCM302x processor family.

- Additional utilities. These components are authored by Analog Devices, and assist in the generation of applications for the ADuCM302x processor family.
- Documentation.

6.6 Source files for device family drivers

ADuCM302x.h	Device descriptions and macro files
System	Source and include files
Startup	Source and include files

Various peripheral device driver sources and include files in “Source” and “Include” directories.

6.6.1 Location

The ADuCM302x Device Family Pack 4.0.0 will be installed into the CMSIS pack directory for the targeted development environment:

Keil uVision	`<keil_root>\Arm\Packs\AnalogDevices\ADuCM302x_DFP\4.0.0`
CCES	`<cces_root>\ARM\packs\AnalogDevices\ADuCM302x_DFP\4.0.0`
IAR Embedded Workbench	`<iar_packrepo>\AnalogDevices\ADuCM302x_DFP\4.0.0`

with

- `<keil_root>`
  - The location where Keil uVision is installed e.g. **`C:\Keil_v5`**.
- `<cces_root>`
  - The location where CrossCOre Embedded Studio is installed, e.g. **`C:\analog\CCES\3.0.3.`**
- `<iar_packrepo>`
  - The location where IAR Embedded Workbench installs CMSIS packs, e.g. **`C:\Users\<windows_username>\AppData\local\IAR Embedded Workbench\PackRepo`**.

6.6.2 Device Driver Thread Safety

All Device Drivers are **not** thread-safe. They are re-entrant but not thread-safe. If an RTOS is present, then drivers will use the RTOS semaphores for implementing the blocking calls.

6.6.3 Contacting Technical Support

You can reach Analog Devices software and tools technical support in the following ways:

- Post your questions in the software and development tools support community at EngineerZone®.
- E-mail your questions about processors and processor applications to processor.support@analog.com.
- For Greater China, Processors and DSP applications and processor questions can be sent to: processor.china@analog.com.
- Submit your questions to technical support directly via http://www.analog.com/support.
- Contact your Analog Devices sales office or authorized distributor.

6.6.4 Examples

This ADuCM302x Device Family Pack comes with a very simple example which requires multiple drivers (DMA, UART, Power)

Examples for drivers

1.	HelloWorld	• Demonstrate how to create a simple application that prints "Hello, world!". • Add LED code to the Hello World example to confirm execution when console output is not available with DAP debugging

6.7 Known Issues

For the latest anomalies please consult our Software and Tools Anomalies Search page.