EB00927

eQADC - Avoiding Unintended Result Swap

Rev. 0 — 02 December 2021

Engineering Bulletin

Contents

1 Issue description

Improper ordering of eQADC commands may lead to unexpected result swap in result FIFO. As result FIFO does not contain any flag or other way how to identify which results are read, it is needed to guarantee ordering of results is the same as the ordering of commands.

Possible result swap reasons are caused by eQADC architecture along with incorrect SW setup. As it is often not clear to users, the following analysis has been prepared.

The document is related to all eQADC implementations excluding the MPC551x family, its eQADC implements only one analog-to-digital converter (ADC0).

NOTE

Knowledge of the eQADC module is expected. Always consult with the latest reference manual for the given device.

NOTE

All figures are simplified, not showing complete ADC timing. They only have to highlight the described issue.

1.1 Swap caused by two-stage command buffers

One reason can be caused by a two-stage command buffer that is used with every ADCs (in the reference manual marked as `CBuffer0` and `CBuffer1` - refer to “CFIFO Prioritization Logic” figure, find it in every device’s RM equipped with eQADC).

In case when one buffer is full and the second one is empty, buffered conversion may result in a swap of the result.

For example, there is the following command queue:

Table 1. Command queue

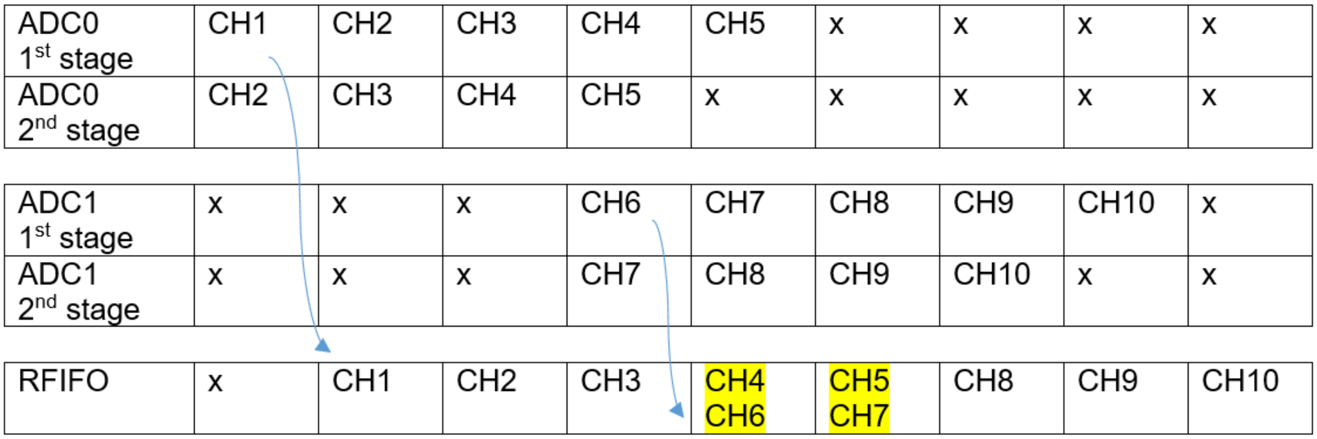
Ordinal number	Command
Command 1	CH1, ADC0
Command 2	CH2, ADC0
Command 3	CH3, ADC0
Command 4	CH4, ADC0
Command 5	CH5, ADC0
Command 6	CH6, ADC1
Command 7	CH7, ADC1

Table continues on the next page...

Table 1. Command queue (continued)

Ordinal number	Command
Command 8	CH8, ADC1
Command 9	CH9, ADC1
Command 10	CH10, ADC1

The diagram below shows filling of eQADC command buffers and converted results stored in result FIFO. Beats given by ADC clock lies on the X-axis. Result swap is highlighted.



ADC0 1st stage	CH1	CH2	CH3	CH4	CH5	x	x	x	x
ADC0 2nd stage	CH2	CH3	CH4	CH5	x	x	x	x	x
ADC1 1st stage	x	x	x	CH6	CH7	CH8	CH9	CH10	x
ADC1 2nd stage	x	x	x	CH7	CH8	CH9	CH10	x	x
RFIFO	x	CH1	CH2	CH3	CH4 CH6	CH5 CH7	CH8	CH9	CH10

Figure 1. RFIFOx filling – first use case

1.2 Swap caused by ADC0 priority in result buffer

The reference manual says:

“In general received data is moved into RFIFOs as they become available, while an exception happens when multiple results from different sources become available at the same time. In that case, result data from ADC0 is processed first, result data from ADC1 is the only process after all ADC0 data is processed, and finally returned data from the companion module is processed (after all data from ADC0/1 is processed).”

If the sequence is as follows, the resulting swap is present as well.

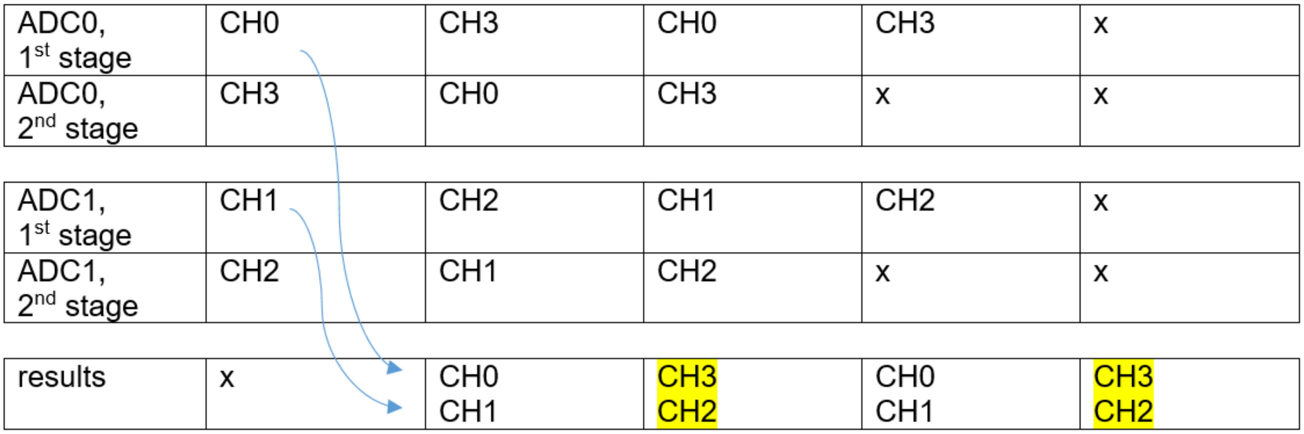
Table 2. Command queue

Ordinal number	Command
Command 1	CH0, ADC0
Command 2	CH1, ADC1
Command 3	CH2, ADC1
Command 4	CH3, ADC0
Command 5	CH0, ADC0

Table continues on the next page...

Table 2. Command queue (continued)

Ordinal number	Command
Command 6	CH1, ADC1
Command 7	CH2, ADC1
Command 8	CH3, ADC0



ADC0, 1st stage	CH0	CH3	CH0	CH3	x
ADC0, 2nd stage	CH3	CH0	CH3	x	x
ADC1, 1st stage	CH1	CH2	CH1	CH2	x
ADC1, 2nd stage	CH2	CH1	CH2	x	x
results	x	CH0 CH1	CH3 CH2	CH0 CH1	CH3 CH2

Figure 2. RFIFOx filling – second use case

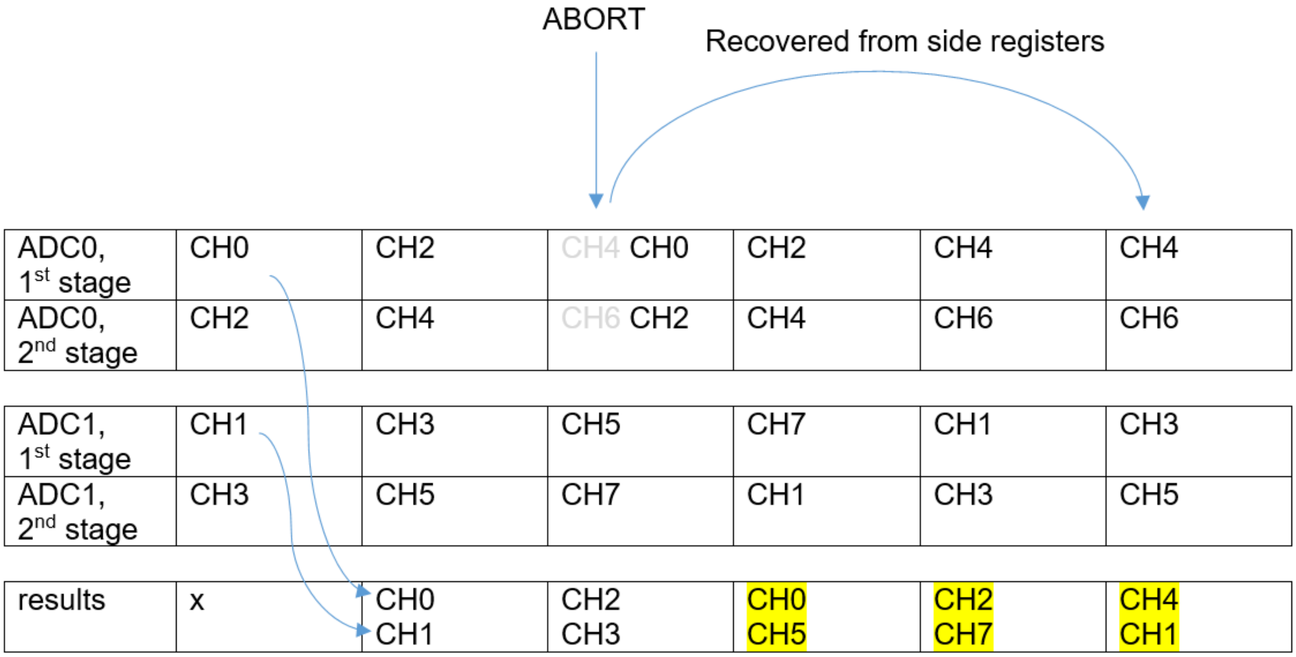
1.3 Swap caused by ABORTing of command in a queue execution

Improper use of the ABORT feature may cause shuffling the results out of the expected order.

In the example configuration below, let us consider ABORT feature is enabled for `CBuffer0` (`EQADC_MCR[ICEA0]` = 1). Triggering of command queue causes abortion of currently executed conversion by ADC0 and these commands are executed after finishing of CFIFO0 execution.

Table 3. Command queue

Ordinal number	Command
Command 1	CH0, ADC0
Command 2	CH1, ADC1
Command 3	CH2, ADC0
Command 4	CH3, ADC1
Command 5	CH4, ADC0
Command 6	CH5, ADC1
Command 7	CH6, ADC0
Command 8	CH7, ADC1



ADC0, 1st stage	CH0	CH2	CH4 CH0	CH2	CH4	CH4
ADC0, 2nd stage	CH2	CH4	CH6 CH2	CH4	CH6	CH6
ADC1, 1st stage	CH1	CH3	CH5	CH7	CH1	CH3
ADC1, 2nd stage	CH3	CH5	CH7	CH1	CH3	CH5
results	x	CH0 CH1	CH2 CH3	CH0 CH5	CH2 CH7	CH4 CH1

Figure 3. RFIFOx filling – third use case

1.4 Swap caused by ERR005086

On certain devices, there is ERR005086 (or e5086). Its indirect implication maybe result in “swap”.The commas as in are not actually swap, but it may look like this in certain configurations.

Errata wording is following:

e5086: eQADC: unexpected result may be pushed when Immediate Conversion Command is enabled

Description:

In the enhanced Queued Analog to Digital Converter (eQADC), when the Immediate Conversion Command is enabled (`ICEAn=1`) in the eQADC_MCR (Module Configuration Register), if a conversion from Command First-In-First Out (`CFIFO0`, conv0) is requested concurrently with the end-of conversion from another, lower priority conversion (convx), the result of the convx may be lost or duplicated causing an unexpected number of results in the FIFO (too few or too many).

Workarounds:

1) Do not use the abort feature (`ICEAn=0`).

2) Arrange the timing of the `CFIFO0` trigger such that it does not assert the trigger at the end of another, lower priority conversion.

3) Detect the extra or missing conversion result by checking the `EQADC_CFTCRx` (`EQADC CFIFO` Transfer Counter Register x). This register records how many commands were issued, so it can be used to check that the expected number of results have been received.

2 Possible solutions

1) Use two command queues and two results queues (therefore two different return tags).

2) Use one command queue and two results queues by using two different return tags.

3) If forced to use one results queue, add three dummy conversions with null return tags to keep `ADC1` busy until `ADC0` returns the last result.

4) Keep regular alternating of commands assigned to `ADC0` and `ADC1` (The ABORT feature must be disabled).

5) Use ABORT functionality with caution.

3 Revision history

Table 4. Revision history

Revision number	Date	Substantive changes
0	02 December 2021	Initial release

How To Reach Us

Home Page:

nxp.com

Web Support:

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.

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) 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, μVision, 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. M, M Mobileye and other Mobileye trademarks or logos appearing herein are trademarks of Mobileye Vision Technologies Ltd. in the United States, the EU and/or other jurisdictions.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 02 December 2021
Document identifier: EB00927