AN14767

ADC Sequencing Using Wait for Trigger Functionality for the i.MX RT118x

Rev. 1.0 — 28 July 2025

Application note

Document information

Information	Content
Keywords	AN14767, i.MX RT118x, ADC sequencing, wait for trigger, ADC triggering, core offload, optimization
Abstract	The analog-to-digital converter (ADC) module incorporates a linear successive approximation algorithm and supports both differential and single-ended operations, offering resolutions of 16 bits or 13 bits for differential mode and 16 bits or 12 bits for single-ended mode. The wait for trigger functionality allows a single sequence to be divided into multiple smaller sequences, enabling different sequences to be started by a single trigger without requiring core intervention.

1 ADC

The analog-to-digital converter (ADC) module incorporates the following:

- A linear successive approximation algorithm
- Supports both differential and single-ended operations
- Supports resolutions of 16 bits or 13 bits for differential mode and 16 bits or 12 bits for single-ended mode

One ADC can handle two simultaneous single-ended conversions. It features channel scaling. It allows input voltage levels to exceed the ADC reference voltage. It also provides configurable analog input sample times and speed options to support low-power modes in system-on-chip (SoC) designs. The module includes a trigger detection system with up to eight trigger sources. Each trigger source is configurable with priority levels and selectable for software or hardware triggering. You can independently configure options and scan channel sequences with 15 command buffers. It supports automatic comparison functions for conditions such as less-than, greater-than, within range, or out-of-range, with options to store results when true or repeat until true. The ADC includes two independent result FIFOs. Each FIFO has 16 entries. Both the FIFOs offer configurable watermarks and overflow detection. Also, the module supports interrupt, direct memory access (DMA), or polled operation modes. It includes calibration logic for linearity and gain adjustment.

2 Trigger

The ADC command execution can be initiated from eight trigger sources. Each trigger can be software generated by writing 0b1 to the corresponding SWTRIG[SWTn] bit field. Alternatively, hardware triggers can be generated from asynchronous input sources at the periphery of the module. The number and sources of hardware triggers implemented is device-specific.

To enable Each hardware trigger source, set the associated enable bit (TCTRLa[HTEN]). Each trigger source is assigned a priority via the associated priority control field (TCTRLa[TPRI]). Each of the trigger sources is associated with a command buffer via the associated command select field (TCTRLa[TCMD]).

The ADC has eight trigger sources and the setting is done in TRIG0 to TRIG7 registers. If you want to use a hardware trigger, it is necessary to set the HTEN bit in the Trigger Control Register. The ADC trigger sources correspond to the respective output signal of XBAR1 and XBAR_OUT140 to XBAR_OUT147. These trigger sources are common for the ADC1 and ADC2. The XBAR1 has a total of 215 inputs. The signal can originate from an on-chip peripheral (eFlexPWM, LPIT timer, and so on) and from an off-chip source via GPIO pin. For more details on the XBAR1 multiplexing, see the *i.MX RT1180 Reference Manual* (document IMXRT1180RM).

3 Wait for trigger functionality

The command sequence is executed as per the order the CMDHa[NEXT] field gives.

Trigger runs the first CMD1. If this command has a pointer to the next command in the CMDHa[NEXT] field, then after CMD1 is done the channel from NEXT field runs automatically. If NEXT filed equals zero, then the sequence is ended.

The CMDHa[WAIT_TRIG] bit enables to “freeze” execution of the sequence until the next trigger event occurs. If a single trigger source triggers the sequence, it is possible to divide one sequence into more sequences.

4 Command buffer register

The command buffer register is 64-bit wide (32-bit CMDL register and 32-bit CMDH register) and each register corresponds to the respective command, CMD1 to CMD15. The settings of the command buffer completely control corresponding sampling, you can set channel, sample time, and so on.

5 Use case

In this use case, a single trigger signal is used, which is a PWM signal acting as the trigger. The goal is to initiate different sequences of the ADC measurements across three consecutive PWM periods. This means we have one major sequence composed of three minor sequences.

It can be implemented either via software or using eDMA. After each minor sequence, the ADC settings are updated so that a different sequence is triggered on the next PWM signal. This approach is suitable for applications where there are no strict constraints on core processing load or eDMA bandwidth.

The objective is to create a mechanism that automatically switches between sequences using only the hardware resources of the ADC peripheral.

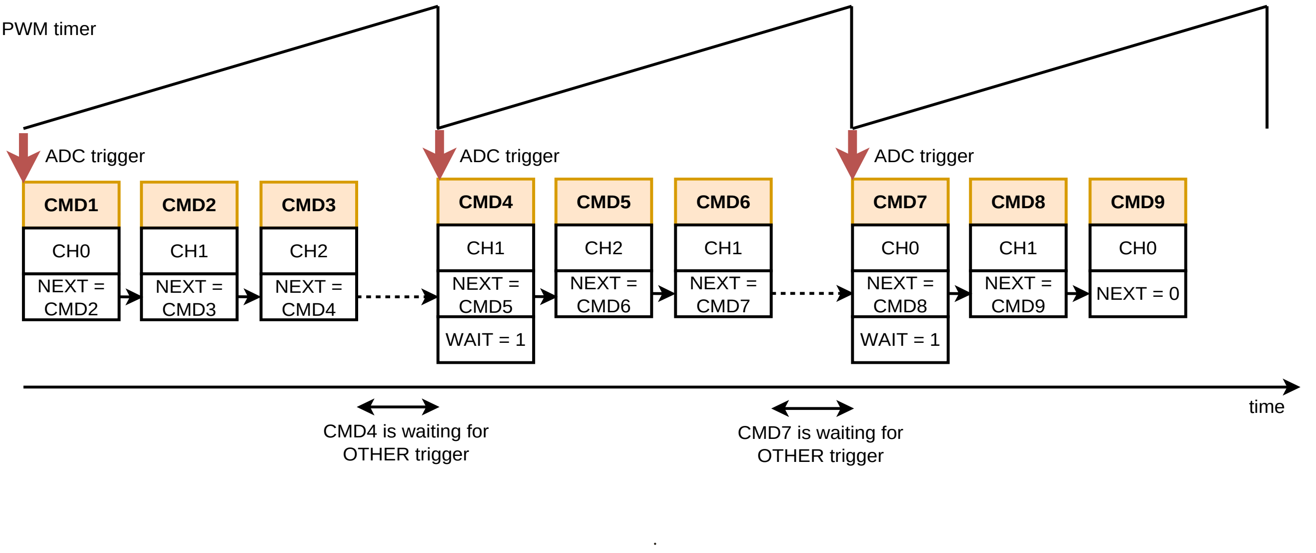

Figure 1. Wait for trigger functionality

6 Code example

First, set the XBAR1 to propagate the trigger from eFlexPWM peripheral to ADC as a trigger source:

```
XBAR_Init(kXBAR_DSC1); // Init XBAR1 peripheral
XBAR_SetSignalsConnection(kXBAR1_InputFlexpwm1Pwm0OutTrig0,
kXBAR1_OutputAdc12HwTrig0); // Set XBAR1 multiplexer
void initADC()
{
ADC1->CFG = ADC_CFG_PWREN_MASK; // ADC analog are pre-enabled
ADC1->TCTRL[0] = ADC_TCTRL_TCMD(1); // Trigger CMD1 and enable trigger
// Sequence 1
for(int i=0; i<9; i++)
{
ADC1->CMD[i].CMDH = ADC_CMDH_NEXT(i+2);
}
ADC1->CMD[3].CMDH |= ADC_CMDH_WAIT_TRIG_MASK;
ADC1->CMD[6].CMDH |= ADC_CMDH_WAIT_TRIG_MASK;
}
```

Set up the eFlexPWM peripheral. For this use case, no specific configuration is required, perform the basic initialization of the eFlexPWM module and set it to a 50 % duty cycle. This setting is configured in the SM0TCTRL register.

Choose the trigger source - the PWM output signal or a signal from a comparator. In this example, the VAL1 compares match as the trigger source is used for the ADC.

```
void initPWM().
{
/****** PWM init *****/
/* Set the PWM Fault inputs to a low value */
XBAR_SetSignalsConnection(kXBAR1_InputLogicHigh, kXBAR1_OutputFlexpwm1Fault0);
XBAR_SetSignalsConnection(kXBAR1_InputLogicHigh, kXBAR1_OutputFlexpwm1Fault1);
XBAR_SetSignalsConnection(kXBAR1_InputLogicHigh,
 kXBAR1_OutputFlexpwm1234Fault2);
XBAR_SetSignalsConnection(kXBAR1_InputLogicHigh,
 kXBAR1_OutputFlexpwm1234Fault3);
XBAR_SetSignalsConnection(kXBAR1_InputFlexpwm1Pwm0OutTrig0,
 kXBAR1_OutputAdc12HwTrig0);
PWM_GetDefaultConfig(&pwmConfig);
#ifdef DEMO_PWM_CLOCK_DEVIDER
pwmConfig.prescale = DEMO_PWM_CLOCK_DEVIDER;
#endif
/* Use full cycle reload */
pwmConfig.reloadLogic = kPWM_ReloadPwmFullCycle;
/* PWM A & PWM B form a complementary PWM pair */
pwmConfig.pairOperation = kPWM_ComplementaryPwmB;
pwmConfig.enableDebugMode = false;
/* Initialize submodule 0 */
if (PWM_Init(BOARD_PWM_BASEADDR, kPWM_Module_0, &pwmConfig) == kStatus_Fail)
{
return 1;
}
PWM_FaultDefaultConfig(&faultConfig);
/* Sets up the PWM fault protection */
PWM_SetupFaults(BOARD_PWM_BASEADDR, kPWM_Fault_0, &faultConfig);
PWM_SetupFaults(BOARD_PWM_BASEADDR, kPWM_Fault_1, &faultConfig);
PWM_SetupFaults(BOARD_PWM_BASEADDR, kPWM_Fault_2, &faultConfig);
PWM_SetupFaults(BOARD_PWM_BASEADDR, kPWM_Fault_3, &faultConfig);
/* Set PWM fault disable mapping for submodule 0/1/2 */
PWM_SetupFaultDisableMap(BOARD_PWM_BASEADDR, kPWM_Module_0, kPWM_PwmA,
 kPWM_faultchannel_0,
kPWM_FaultDisable_0 | kPWM_FaultDisable_1 | kPWM_FaultDisable_2 |
 kPWM_FaultDisable_3);
PWM1->SM[0].TCTRL = PWM_TCTRL_OUT_TRIG_EN(0x2); // Enable output trigger when
 counter value matches the VAL1
/* Call the init function with demo configuration */
PWM_InitSignals();
/* Set the load okay bit for all submodules to load registers from their buffer
 */
PWM_SetPwmLdok(BOARD_PWM_BASEADDR, kPWM_Control_Module_0, true);
BOARD_PWM_BASEADDR->SM[0].DMAEN |= PWM_DMAEN_CAPTDE(0x2); // A local
 synchronization (VAL1 matches counter) sets the read DMA request
/* Start the PWM generation from Submodules 0, 1 and 2 */
PWM_StartTimer(BOARD_PWM_BASEADDR, kPWM_Control_Module_0);
}
```

7 Abbreviations

Table 1 lists the acronyms used in this document.

Table 1. Acronym and abbreviations

Acronym	Description
ADC	Analog-to-digital converter
DMA	Direct memory access
FIFO	First in first out
SoC	System-on-chip

8 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

9 Revision history

Table 2. Revision history

Document ID	Release date	Description
AN14767 v.1.0	28 July 2025	Initial public release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. 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 B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Figures

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.