Freescale Semiconductor, Inc.
Application Note

Document Number: AN5168
Rev. 0, 08/2015

Using the ADC Module in S12ZVM

By: Agustin Diaz

1. Introduction

This application note provides introduction to the Analog-to-Digital Converter module of the S12ZVM family. The ADC12B_LBA is a successive approximation analog-to-digital converter with a maximum of N internal channel. This module has List Based Architecture (LBA) which allows the user to have a free selection of the conversion channel and the flexibility to have different conversion sequences. For more information see the ADC chapter of the reference manual at www.freescale.com.

Contents

1. Introduction 1
2. The ADC12B_LBA module 2
   2.1. Initialization 2
   2.2. Calculating the ADC Clock 3
   2.3. Modes of Operation (Flow Control) 3
   2.4. Command List and Result List Offset 3
   2.5. The Command List 4
   2.6. Double Buffer/Single Buffer List 5
3. Attachments 7

2. The ADC12B_LBA module

The ADC12B_LBA is a n-channel multiplexed input successive approximation analog-to-digital converter.

The List Based Architecture (LBA) provides flexible conversion sequence definition as well as flexible oversampling. The order of channels to be converted can be freely defined. Also, multiple instantiations of the module can be triggered simultaneously (matching sampling point across multiple module instantiations).

The module offers two ways of controlling the flow of the conversions:

- Via data bus accesses
- Via internal interface signal

The following is a summary of the most notable ADC12B_LBA module features:

- List Based Architecture for conversion command and result value organization
- Selectable resolution of 8-bit, 10-bit, 12-bit
- Channel select control for N external analog input channels
- Provides internal device channels such as: temperature sensor, bandgap voltage, GDU phase multiplexer voltage, GDU DC link voltage.
- Programmable sample time
- Left/right justified result data
- Reference voltage selection
- 15 conversion interrupts
- Dedicated interrupt for “End Of List” commands
- Command Sequence List with a maximum number of 64 commands entries

2.1. Initialization

To use the ADC module follow the steps shown below:

- Configure the access mode: Null Access, Internal Interface, Data Bus, Dual Access (ADCCTL_0_ACC_CFG)
- Configure the flow control (ADCCTL_0_MOD_CFG)
- Select the buffer mode for Command List: Single, Dual (ADCCTL_1_CSL_BMOD)
- Select the buffer mode for Result List: Single, Dual (ADCCTL_1_RVL_BMOD)
- Select the data justification: Left or Right (AD0FMT_DJM)
- Select the bit resolution: 8 bit, 10 bit, 16 bit (ADCFMT_RES)
- Select the frequency for the ADC clock

- Establish the base pointer for the Command List and Result List (ADCCBP and ADCRBP)
- Establish the Command/Result offset registers
- Enable the interruptions if needed (ADCCONIE and ADCEIE)
- Enable the ADC module (ADCCTL_ADC_EN)
- It is recommended to issue a restart event and wait for the restart flag to be cleared

2.2. Calculating the ADC Clock

The register that controls the ADC Clock is the ADCTIM and bits are the PRS[0:6].

The formula is:

$$PRS[0:6] = \frac{f_{bus}}{2 * f_{ATDCLK}} - 1$$

Where f_{ATDCLK} must be between 0.25 and 8 MHz.

2.3. Modes of Operation (Flow Control)

The mode of operation is controlled by the ADCCTL_0 register and the MOD_CFG bits. These bits defines the conversion flow control after a Restart Event or “End of List” type command. There are two modes of operation: Trigger Mode and Restart Mode.

In **Trigger Mode** after the “End of List” type command the command list returns to the top of the commands and waits for trigger event to start again.

In **Restart Mode** after the “End of List” type command a restart event must be issued in order to restart the list to the top of the commands. Then a trigger event must happened in order to the list to flows. It is important to wait for the restart event flag to clear before issuing a trigger event or an error will be set.

2.4. Command List and Result List Offset

When using a double buffer mode for any of the lists an offset must be set in order to separate one from the other. The offsets registers are ADCCROFF0 and ADCROFF1.

These bits do not represent absolute addresses instead it is a sample offset.

- For Result List the object size is 16 bits
- For the Command List the object size is 64 bits

2.5. The Command List

The command list is where the configuration of the conversions are store. It contains the type of command, corresponding interruption, reference voltage, selected channel and sample time. These commands are controlled by the registers: ADCCMD_0, ADCCMD_1 and ADCCMD_2.

The next table illustrates how the command list is build:

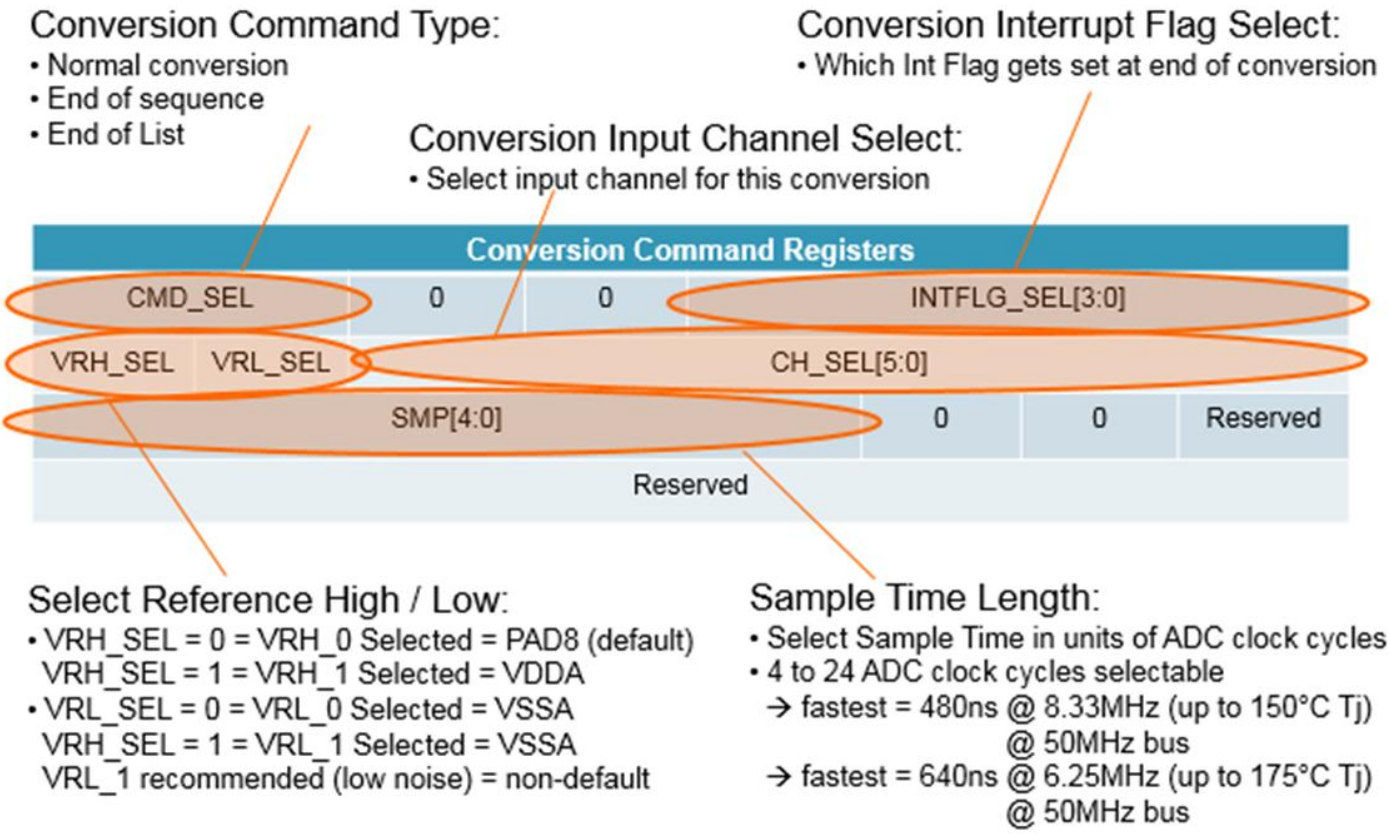


Figure 1. Command list

2.6. Double Buffer/Single Buffer List

The ADC module allows the programmer to have two command lists and two result lists. The bits that control the number of lists are CSL_BMOD and RVL_BMOD. These lists can interact with each other in four different ways.

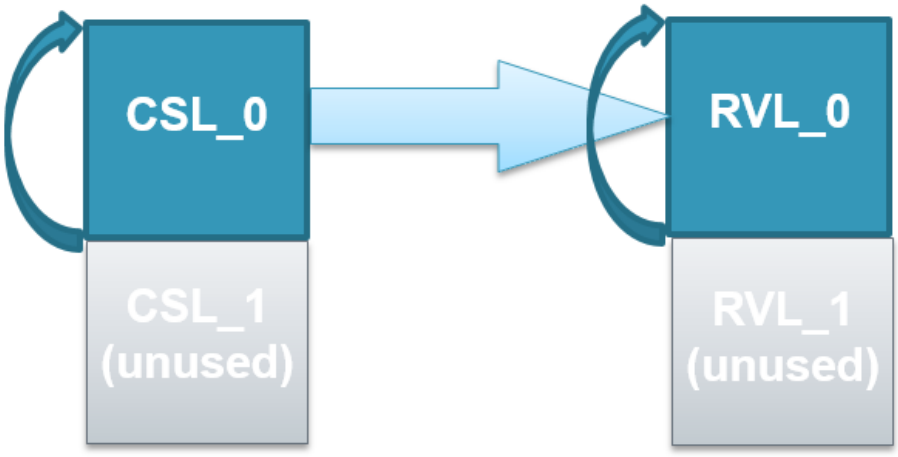


Figure 2. CSL Single Buffer Mode – RVL Single Buffer Mode

After an "end of list" command CSL and RVL indexes reset for both the CSL_0 and the RVL_0.

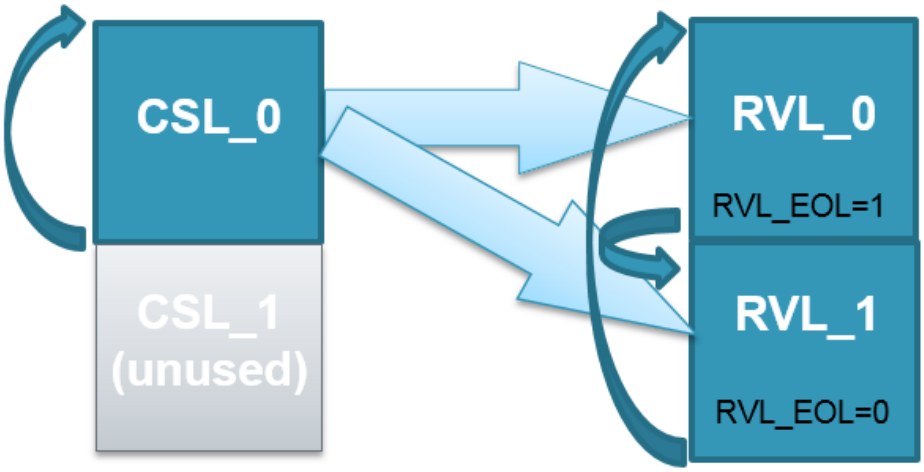


Figure 3. CSL Single Buffer Mode – RVL Double Buffer Mode

After an "end of list" command executed the RVL lists change between each other. The RVL_EOL bit shows the last active RVL when "End of list" command was executed.

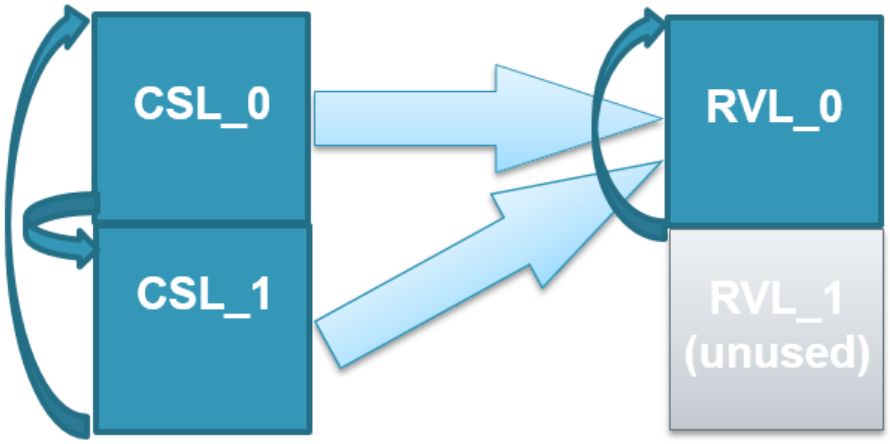


Figure 4. CSL Double Buffer Mode – RVL Single Buffer Mode

After an “end of list” command in order to change from one CSL to the other, the bits RSTA and LDOK from ADCFLWCTL register must be set simultaneously. After “end of list” command the RVL index restarts.

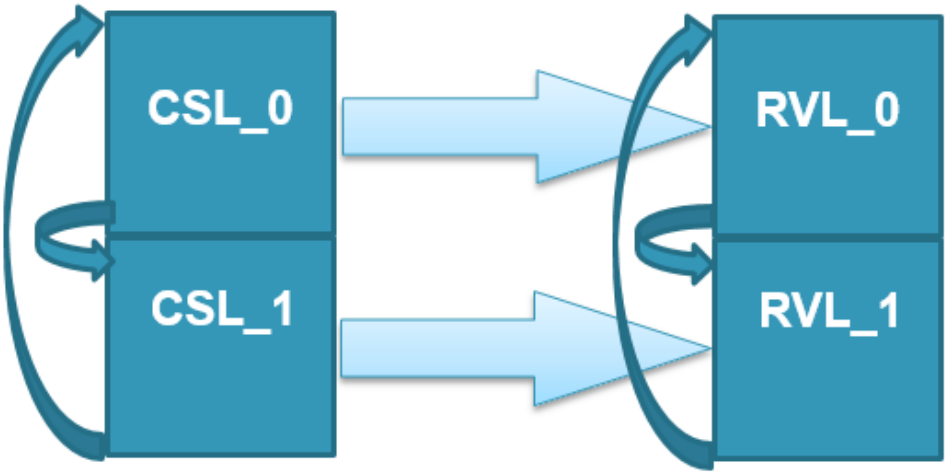
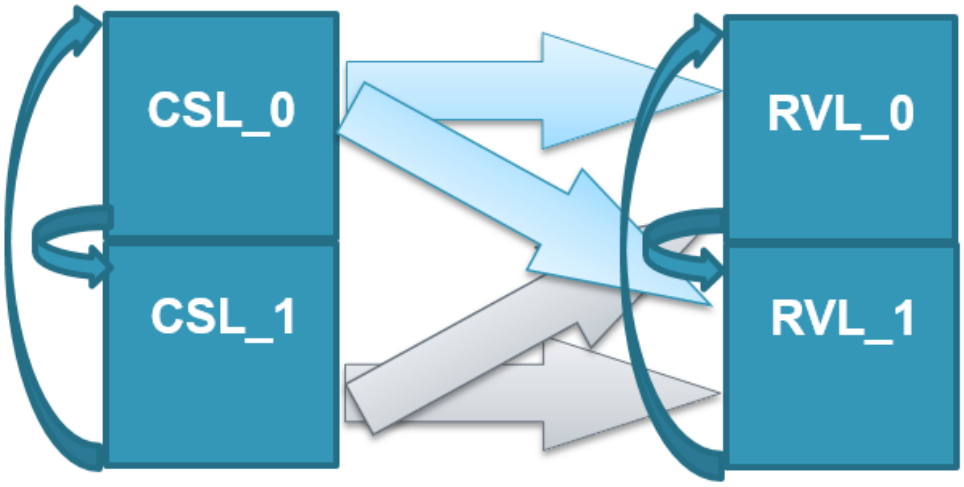


Figure 5. CSL Double Buffer Mode – RVL Single Buffer Mode

After every “end of list” command the bits RSTA and LDOK must the set simultaneously, in order to change from one CSL to the other. The “end of list” command will also change the RVL from one to the other. CSL_0 will be related to RVL_0 and CSL_1 to RVL_1.



End of list command does not necessarily mean to change between CSL_0 to CSL_1. The bit LDOK does not necessary set simultaneously with RSTA. The RVL change between RVL_0 to RVL_1 at every “end of list” command or aborted CSL.

3. Attachments

This application note includes the software that carries out the initialization described in initialization. It also includes example for each of the configuration explained in Double Buffer/Single Buffer.

- CSL Single Buffer / RVL Single Buffer
- CSL Single Buffer / RVL Double Buffer
- CSL Double Buffer / RVL Single Buffer
- CSL Double Buffer / RVL Double Buffer

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: AN5168
Rev. 0
08/2015