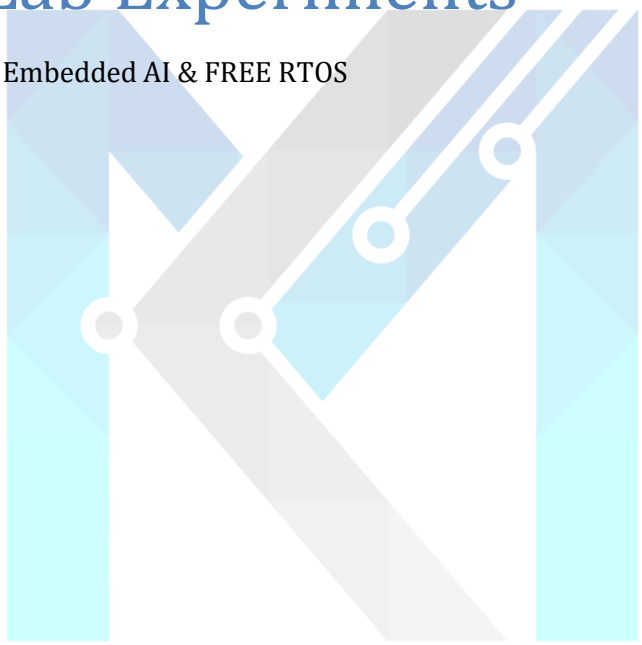
Kernel Masters

ADC Lab Experiments

Embedded C , Embedded AI & FREE RTOS



Kernel Masters
12/16/2023

Contents

Introduction to Analog to Digital Converter (ADC)

The 12-bit ADC is a successive approximation analog-to-digital converter. It has up to 19 multiplexed channels allowing it to measure signals from 16 external sources, two internal sources, and the VBAT channel. The A/D conversion of the channels can be performed in single, continuous, scan or discontinuous mode. The result of the ADC is stored into a left or right-aligned 16-bit data register.

The analog watchdog feature allows the application to detect if the input voltage goes beyond the user-defined, higher or lower thresholds.

Single Conversion Mode:

In single conversion mode, the ADC performs a single conversion when triggered.

The ADC is triggered by a software command or an external event, and it converts one channel's analog input to a digital value.

After the conversion is complete, the ADC stops, and the result is typically available for reading from the ADC data register.

Single conversion mode is useful when you only need occasional ADC measurements and want to minimize power consumption.

Continuous Conversion Mode:

In continuous conversion mode, the ADC continuously performs conversions without the need for external triggering.

Once configured in continuous mode, the ADC automatically starts a new conversion as soon as the previous one is complete.

The results are usually stored in a data register, and you can continuously read the latest conversion results from this register.

Continuous conversion mode is beneficial when you need a continuous stream of ADC measurements, such as in applications where you are continuously monitoring an analog signal.

Continuous or Single Scan:

The scan mode can be configured to operate in a continuous or single-scan mode.

In continuous scan mode, the ADC continuously cycles through the specified channel sequence.

In single-scan mode, the ADC completes one full sequence of conversions and then stops.

Injected Chanel vs Regular channel:

You can configure the ADC to read in a sequence of channels in a loop. Those channels are being converted regularly. In injected mode conversion is triggered by an external event or by software. An injected conversion has higher priority in comparison to a "regular" conversion and thus interrupts the regular conversions.

The injected mode is a mode where the ADC conversion can be "injected" during the conversion of regular channels due to some trigger (timer or something else). This is useful, for example in motor control application, to delay conversion until after some event is complete (such as transistor switching) so that the conversion noise is reduced.

ADC Lab Experiments:

ADC Controller

The Raayan Mini Development board has two on-board analog devices which is one LM35 temperature sensor and one potentiometer. These devices are internally connected to ADC1_10 and ADC1_11 analog channels respectively. To convert the sensor output voltage into a digital value, one 12-bit analog-to-digital converter is embedded and shares up to 16 external channels, performing conversions in the single shot or scan mode.

GPIO Pin	Pin Function	On-Board Function	Device
PC0	GPIO	ADC1_IN10	LM35 Sensor
PC1	GPIO	ADC1_IN11	Potentiometer

1. Lab Experiment 1 [SINGLE CONVERSION MODE]:

Write an Embedded C program to initialize ADC Single Conversion mode. Using regular channel software trigger samples ADC channel 11 (potentiometer) and stores the result to a global variable ADC that can be accessed with the ST-Link Debugger.

a) Polling Method

Pseudo code:

ADC Initialization:

- Set Port C bit in RCC_AHB1ENR register to enable port C clock.
- Load “11” in to PC1 bit filed in GPIOC_MODE Register.
- Set ADC clock bit in RCC_APB2ENR to enable ADC clock.
- Load “1011” (channel no 11) in to 4-0 bit fields in ADC_SQR3 register.
- Set 10th bit (EOCS) in ADC_CR2 to select End of Conversion.
- Set 0th bit ADC_CR2 to enable ADC.

ADC Operation:

Super loop:

- Set 30th bit (SWSTART) in ADC_CR2 register to start regular channel software trigger.

Loop:

- Read ADC_SR and check 1st bit (EOC), if bit is set go to next statement otherwise go to loop.
- Read ADC_DR register and write in to global variable.
- Goto Super loop.

b) Interrupt Method

Interrupt Initialization,
- Enable EOCIE bit in ADC_CR1 register to enable End Of Conversion interrupt.
- Enable 18th bit in NVIC_ISER0 to enable ADC interrupt in NVIC.

In ADC ISR,
- Read ADC_DR register and write in to global variable.
- To clear EOC bit in ADC_SR to give EOC interrupt acknowledge and wait for another EOC interrupt.

2. Lab Experiment 2: [SCAN MODE]

Write an Embedded C program to initialize ADC Scan mode.
Using regular channel software trigger samples ADC channel 10 (Temperature sensor) and 11 (potentiometer) and stores the result to a global variable that can be accessed with the ST-Link Debugger. using interrupt method.

Hint: To read multiple channels, no. of channels mentioned in length field in ADX_SQR1 register.
In this example to read two channels(10,11) so length is 2.
1st channel(10) load in to SQR3 in SQ1 bit fields and
2nd channel(11) load in to SQR3 in SQ2 bit fields.

3. Lab Experiment 3: [ADC WATCHDOG]

Write an Embedded C program to initialize ADC Scan mode.
load lower threshold value 1000 in to ADC_LTR and higher threshold value 2000 in to ADC_HTR. Enable ADC watchdog timer interrupt.

Test case: Assign a break point inside ADC ISR, Adjust potentiometer knob, whenever to reach beyond threshold values program stop in ISR.

Using regular channel software trigger samples ADC channel 11 (potentiometer) and stores the result to a global variable that can be accessed with the ST-Link Debugger. using interrupt method.