**Lab 3: Analog to Digital Converter (ADC)**

**Instructor: Prof. Yifeng Zhu**

**Fall 2019**

## Goals

1. Use software trigger to make ADC conversions, and polling method to read ADC results
2. Understand the concept of ADC resolution
3. Print out the battery voltage by using the USART driver made in last lab.

## Pre-lab Assignment

1. Read Chapter 20 Analog to Digital Conversion (ADC)
2. Complete the pre-lab assignment

## Lab Demo

1. Implement ADC1\_Init().
2. Print out the battery voltage out on a serial terminal.

## Reference Voltage

The reference voltage to the analog-to-digital (ADC) and digital- to-analog (DAC) converters can be provided externally or internally.

* If an ***external*** voltage is selected as the reference voltage, the external reference voltage should be applied to the VREF+ pin.
* If an ***internal*** voltage is selected as the reference voltage, the VREF+ pin can provide the reference voltage for external components.
	+ The internal voltage reference is enabled by setting the ENVR bit in the VREFBUF\_CSR register.
	+ The internal voltage reference can be selected by the VRS bits in the VREFBUF\_CSR register.
		- If VRS is set, the voltage reference is ~2.5V.
		- If VRS is cleared, the voltage reference is ~2.048V.
1. **Using the software to trigger one ADC conversion**:
	1. Software can start one ADC conversion by setting the ADC\_CR2\_ADON, and ADC\_CR2\_SWSTART bit in the ADC1->CR2 register.
	2. The conversion result is saved in register ADC1->DR.

# Lab

1. First you need to set the ADC clock prescaler which is done in the ADC->CCR register. Next set different settings shown in the table below on ADC1. The battery voltage is connected to the analog to digital converter Channel 9. Disable is just setting it to 0U. Initialize the ADC in the ADC1\_Init().

|  |  |
| --- | --- |
| GPIO clock | RCC\_APB2ENR\_ADC1EN |
| GPIO setting | GPIO pin 1 port B |
| GPIO pull | NOPULL |
| GPIO mode | Analog  |
| Clock | ADC\_CLOCKPRESCALER\_PCLK\_DIV4 |
| ADC Scan mode | Disable |
| ADC resolution | 12 bits |
| ADC data alignment | Right |
| External Trigger polarity (EXTERNALTRIGCONVEDGE) | None |
| ADC continuous mode | Disable |
| ADC regular discontinuous mode | Disable |
| Number of conversions  | 1 |
| ADC DMA continuous request (DDS) | Disable |
| ADC end of conversions select(ECOS) | EOC\_SINGLE\_CONV |
| Sampling time | ADC\_SAMPLETIME\_3CYCLES |
| Channel  |  ADC 9 |
| Rank | 1 |

1. The old sample time needs to be cleared and set which is register SMPR2, bits SMP9[2:0]. The rank needs to be set by setting the ADCx\_SQR3 register, bits SQ1[4:0] to ADC\_CHANNEL\_9.

**ECE 271 pre-Lab Assignment**

**Lab 3: Analog to Digital Converter (ADC)**

Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Setting of ADC Control Registers**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Offset** | **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| 0x04 | **ADC\_CCR** | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TSVREFE. | VBATE | Res. | Res. | Res. | Res. | ADCPRE | Res. | Res. | Res. | Res. | Res. |  |  |  |  |  |  |  |  |  |  |  |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x04 | **ADCx\_CR1** | Res. | Res. | Res. | Res. | Res. | OCRIE | RES[1:0] | AWDEN | JSWSTART | Res. | Res. | Res. | Res. | Res. | Res. | DIS NUM [2:0] | JDISCEN | DISCEN | JAUTO | AWD SGL | SCAN | JEOCIE | AEDIE | EOCIE | AEDCH[4:0] |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x08 | **ADCx\_CR2** | Res. | SWSTART | EXTERN[1:0] | EXTSEL [3:0] | Res. | JSWSTART. | JEXTEN [0:1] | JEXTSEL [3:0] | Res. | Res. | Res. | Res. | ALIGN | EOCS | DDS | DMA | Res. | Res. | Res. | Res. | Res. | Res. | CONTINUOUS  | ADON |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x0C | **ADCx\_CFGR** | JQDIS. | AWD1CH[4:0] | JAUTO | JAWD1EN | AWD1EN | AWD1SGL | JQM | JDISCEN | DISCNUM [2:0] | DISCEN | Res. | AUTDLY | CONT | OVRMOD | EXTEN[1:0] | EXTSEL [3:0] | ALIGN | RES [1:0] | Res. | DMACFG | DMAEN |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x2C | **ADCx\_SQR1** | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | L[3:0] | SQ16[4:1] | SQ16\_0 |  SQ15[4:0] | SQ14[4:0] | SQ13[4:0] |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x14 | **ADCx\_SMPR2** |  Res. |  Res. | SMP9[2:0] | SMP8[2:0] | SMP7[2:0] | SMP6[2:0] | SMP5[2:0] | SMP4[2:0] | SMP3[2:0] | SMP2[2:0] | SMP1[2:0] | SMP0[2:0] |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0x18 | **ADCx\_SQR3** | Res. | Res. | SQ6[4:0] | SQ5[4:0] | SQ4[4:1] | SQ4\_0 | SQ3[4:0] | SQ2[5:0] |  SQ1[4:0] |
| Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **Start ADC**
	1. The ADC needs to be turned on, which is register ADON of CR2, and can be done so setting the bit to 1. Also enable (set to 1) the ADC conversion of regular channels which is the register ADC1->CR2 bit ADC\_CR2\_SWSTART.
2. **Calculate the Battery voltage**
	1. The ADC can be read by reading the data register (DR), and it returns a uint32\_t. This value now needs to be converted to the actual value. Since a 12 bit resolution is used, the equation is shown below.

$$V\_{battery}=\frac{ADC value\*V\_{ref}}{2^{12}}\*\frac{R\_{up}+R\_{down}}{R\_{down}}$$

* 1. The boards reference voltage is 3.3V. The battery is connected to the board using a pull up (10kΩ) and a pull down resistor (20kΩ).
	2. Once you calculate the battery voltage print it out on a serial terminal. A charged battery should produce about 4 - 4.20V.

1. **End ADC**
	1. The ADC needs to be turned off, and can be done so setting the ADON bit to 0.

**ECE 271 Lab Demo**

**Lab 3: Analog to Digital Converter (ADC)**