ADC

Except this note,
you also need to look and search anything you need in datasheet.

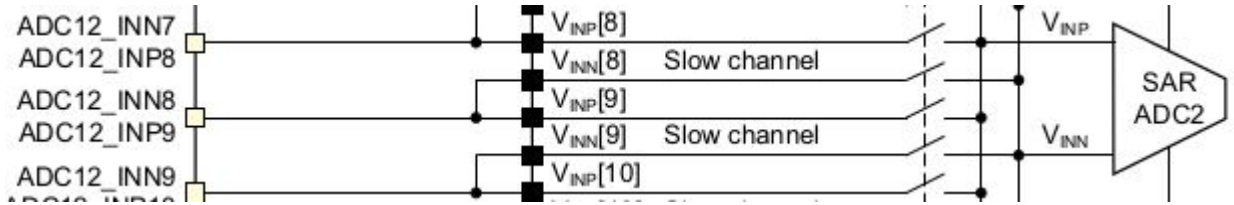
STM32L476

- Up to 3x ADCs, out of which two of them can operate in dual mode
- 12, 10, 8 or 6-bit configurable resolution
- Channel-wise programmable sampling time
- Start-of-conversion can be initiated:
  - by software
  - by hardware triggers with configurable polarity (internal timers events or GPIO input events)
- Conversion modes
  - Each ADC can convert a single channel or can scan a sequence of channels
  - Single mode converts selected inputs once per trigger
  - Continuous mode converts selected inputs continuously

Channel Selection (SQRx)

- There are up to 19 multiplexed channels per ADC
- Each ADC has different channels and same channels both, you can check in RM0351 Reference manual Figure 68. ADC1 connectivity, Figure 69. ADC2 connectivity and Figure 70. ADC3 connectivity

input channel name
it can help you find
correct GPIO



Channel Selection (SQRx) (cont.)

- You can find input channel mapping with GPIO in STM32L476xx Production data Table 16. STM32L476xx pin definitions
- Before any conversion of an input channel coming from GPIO parts, it is necessary to configure the corresponding GPIOx ASCR register in the GPIO, in addition to the I/O configuration in analog mode.

Table 16. STM32L476xx pin definitions (continued)

Pin Number											Pin name (function after reset)	Pin type	I/O structure	Notes	Pin functions	
LQFP64	LQFP64_SMPS	WLCSP72	WLCSP72_SMPS	WLCSP81	LQFP100	UFBGA132	UFBGA132_SMPS	LQFP144	LQFP144_SMPS	UFBGA144					Alternate functions	Additional functions
-	-	-	-	-	-	-	-	20	20	F1	PF8	I/O	FT_a	-	TIM5_CH3, SAI1_SCK_B, EVENTOUT	ADC3_IN11

There has input channel name connected by which pin

Channel Selection (SQRx) (cont.)

- The regular channels and their order in the conversion sequence must be selected in the ADCx_SQR registers. The total number of conversions in the regular group must be written in the L[3:0] bits in the ADCx_SQR1 register.

Channel-wise Programmable Sampling Time (SMPR1, SMPR2)

- Before starting a conversion, the ADC must establish a direct connection between the voltage source under measurement and the embedded sampling capacitor of the ADC.
- Each channel can be sampled with a different sampling time which is programmable using the SMP[2:0] bits in the ADCx_SMPR1 and ADCx_SMPR2 registers.
  - SMP = 000: 2.5 ADC clock cycles
  - SMP = 001: 6.5 ADC clock cycles
  - SMP = 010: 12.5 ADC clock cycles
  - SMP = 011: 24.5 ADC clock cycles
  - SMP = 100: 47.5 ADC clock cycles
  - SMP = 101: 92.5 ADC clock cycles
  - SMP = 110: 247.5 ADC clock cycles
  - SMP = 111: 640.5 ADC clock cycles

Conversion Mode

- In single conversion mode, the ADC performs once all the conversions of the channels.
- In continuous conversion mode, when a software or hardware regular trigger event occurs, the ADC performs once all the regular conversions of the channels and then automatically re-starts and continuously converts each conversions of the sequence.

- Config by using CONT bit in ADCx_CFGR

Programmable Resolution

- It is possible to perform faster conversion by reducing the ADC resolution.
- The resolution can be configured to be either 12, 10, 8, or 6 bits by programming the control bits RES[1:0].

End of Conversion flag and interrupt

- The ADC sets the EOC flag as soon as a new regular conversion data is available in the ADCx_DR register. An interrupt can be generated if bit EOCIE is set. EOC flag is cleared by the software either by writing 1 to it or by reading ADCx_DR.

ADC Startup clock

- check ADC clock source you want, you can find description in RM0351 6.2.13 ADC clock
- also about CKMODE in ADC123_CCR and RCC (clock tree)
- if you don’t enable clock for ADC, you can’t enable ADC

ADC Enable and Start Procedure

- To start ADC operations, it is first needed to exit deep-power-down mode by setting bit DEEPPWD=0.
- It is mandatory to enable the ADC internal voltage regulator by setting the bit ADVREGEN=1 into ADCx_CR register.
- Once DEEPPWD=0 and ADVREGEN=1, the ADC can be enabled
  - Set ADEN=1.
  - Wait ADRDY=1.
  - Regular conversion can then start by setting ADSTART=1

How to get converted data

18.6.15 ADC regular Data Register (ADC_DR)

Address offset: 0x40

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **RDATA[15:0]**: Regular Data converted

These bits are read-only. They contain the conversion result from the last converted regular channel. The data are left- or right-aligned as described in *Section 18.4.26: Data management*.

References

- RM0351 Reference manual, STM32L4x6 advanced ARM®-based 32-bit MCUs
- STM32L476xx, Datasheet