Hello and welcome to this presentation of the STM32H5 Analog-to-Digital Converter or ADC and Digital-to-Analog Converter or DAC.
It will cover the main features of ADC1 and ADC2, which are used to convert analog voltages, like sensor outputs, to digital values for further processing in the digital domain.
The DAC is used to convert digital signals to analog voltages which can interface with the external world and also with on-chip peripherals such as comparator and operational amplifier.

Analog to digital converter (ADC)

Let’s start with a description of the features of the ADCs.

ADC Overview

Features		STM32H503	STM32H56x/573
ADC units		1 module	2 modules
Dual mode operation		-	Yes
Input channels	External (GPIOs)	16	20
	Internal	4	4
Type of conversion		12 bits successive approximation	
Data register		16 bits	
Maximum sampling speed		5 Msps (there is a package dependency)	
Hardware offset calibration		Yes	
Differential Inputs		Yes	
Injected channel conversion		Yes	
Flexible sampling time		Yes	
Oversampling		Up to x256	
Offset compensation		Yes	
Gain compensation		No	

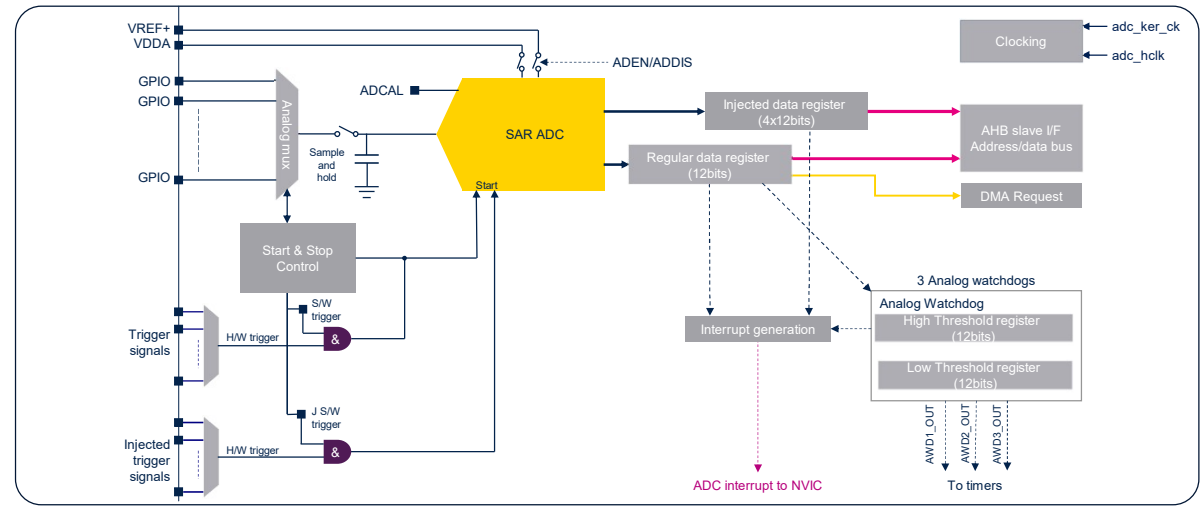
life.augmented

3

Up to two 12-bits analog-to-digital converters are integrated inside STM32H5 products.
The input channel is connected to up to 20 channels capable of converting signals in either Single-end or Differential mode.
Several functional modes are supported; they will be described in the next slides. The ADCs can convert signals at a rate of 5 mega samples per second in 12-bit mode on fast channels in the STM32H56x and STM32H573. The maximum performance for STM32H503 is 2.5 mega samples per second.
All the packages support fast channels, which are ADC inputs 0 to 5. Those channels are designed to have a low resistive analog switch on the signal path.

There are also several different triggering methods.
The injected group can pre-empt the execution of the regular group sampling sequence.
In order to offload the CPU, the ADC has an analog watchdog for monitoring thresholds.
The ADC also offers oversampling to extend the number of bits presented in the final conversion value.
The maximum oversampling ratio is 256.
ADCs conversions are organized in two groups: regular and injected group.
Raw samples acquired by ADC maybe processed by the oversampler and offset compensation units before being provided to the software.
For power-sensitive applications, the ADC offers several low-power features.

Block Diagram



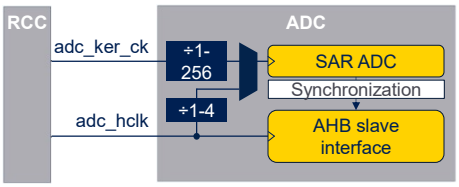
This slide shows the general block diagram of the ADC.
The main important sub-units of the ADC are:

- The power supplies and on/off control
- The analog front-end
- The trigger logic
- The digital back-end including the analog watchdogs and the AHB slave interface
- The clocking.

The next slides detail all these sub-units.

ADC clocking

- The ADC implements a dual clock domain architecture
  - The user can select the root clock used to obtain the ADC clock
    - **Option 1**: adc_ker_ck can be selected by RCC
      - SYSCLK or HCLK
      - pll2_r_ck
      - HSE
      - HSI16
      - CSI
    - **Option 2**: the AHB clock, divided by a programmable factor (1, 2 or 4)



Option	Advantages
1	ADC clock frequency is independent of the AHB clock speed
2	No resynchronization delay, because a unique clock domain is used

life.augmented

5

The ADC has two clock inputs: adc_ker_ck and adc_hclk. The AHB interface belongs to the adc_hclk clock domain. Regarding the digital part of the SAR ADC, there are two options:

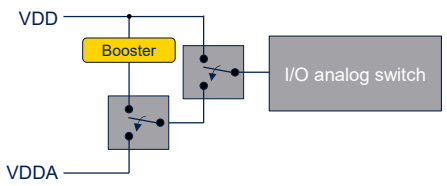
- By using adc_ker_ck, there is no limitation of the CPU nor bus clock speed.
- By using adc_hclk as the reference clock, it is possible to have fixed trigger latency from the timer trigger to the start of ADC conversion.

ADC input channels

	ADC internal channels	ADC external channels connected to dedicated GPIO pads
STM32H503	VREFINT (bandgap voltage =~1.2V) VBAT/4 VSENSE (temperature sensor voltage) Vcore(Digital Core Voltage)	16
STM32H56x/573		20 Analog inputs are connected to both ADC1 and ADC2, except INP2 and INP6/INN2

- IO booster
  - When VDDA voltage is lower than 2.7V, it is necessary to enable the IO booster circuit, that controls the gate voltage of the analog switch in the GPIOs
  - This is activated by SBS_PMCR[BOOSTEN,BOOSTVDDSEL]

VDD	VDDA	BOOSTEN	BOOSTVDDSEL	Gate Voltage
-	> 2.7 V	0	0	VDDA
> 2.7 V	< 2.7 V		1	VDD
< 2.7 V		1	0	Boost Voltage (~3.0V supplied by VDD)



life.augmented

6

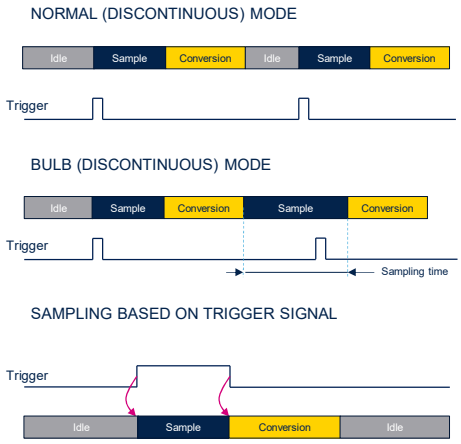
ADCs have the following analog input channels:
- Up to 20 external channels
- 4 internal dedicated channels:
- One channel for internal reference voltage (VREFINT)
- One channel for internal temperature sensor (VSENSE)
- One channel for VBAT monitoring channel (VBAT/4)
- One channel for the internal digital core voltage (VDDCORE)

The analog switch inside the IO has a resistance which increases when the analog switch supply decreases.
For the cases where VDDA and VDD are low, there is a possibility to enable a voltage booster which will supply the analog switch and guarantee low resistance.

It is recommended to use the VDDA supply for the analog. But when VDDA is lower than 2.7 Volts and VDD is larger than 2.7 Volts, the power supply can be switched to VDD. If both VDDA and VDD are lower than 2.7 Volts, the voltage booster should be enabled.

ADC Sampling time control

- Channel-wise programmable sampling time
  - Different sampling times available (2.5, 6.5, 12.5, 24.5, 47.5, 92.5, 247.5, 640.5 cycles)
- Flexible Sampling time
  - Bulb sampling mode
    - Only available in discontinuous mode
    - Sampling starts immediately after last conversion
    - Useful with high impedance sources that require a very long sampling time
  - Sample time control trigger mode
    - Sampling time is fully controlled by the trigger signal
    - Rising edge starts sampling
    - Falling edge stops sampling and the conversion starts



life.augmented

7

This slide describes the sampling time control of ADC.
The first step of a conversion consists in loading the Sample & Hold capacitor with the voltage to be measured.
Longer sample times ensure that signals having a higher impedance are correctly converted.
The sampling times, in ADC clock cycles, are listed on this slide, from 2.5 to 640.5.
The sampling time can be programmed individually for each input channel of ADC.
The ADC implements two other mechanisms to control the sampling time.
The first one is the bulb mode which works only in discontinuous mode.
In this mode, sampling starts immediately after the last

conversion finished without going to idle state. This provides less latency from the trigger signal to the start of the conversion.
The very first ADC conversion, after the ADC is enabled, is performed with the sampling time programmed in SMP bits.
The Bulb mode is effective after the second conversion.
The second mechanism is the sampling mode based on the trigger signal.

- Rising edge starts sampling.
- Falling edge stops sampling and starts the conversion.

Calibration

- ADC offset calibration
  - Offset calibration must be executed at ADC start up and also when VREF or the temperature changes
    - Calibration takes 135 clock cycles
      - Software can copy the calibration factor to flash memory to speed up the start up sequence at the next wake up

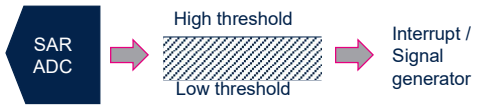
life.augmented

8

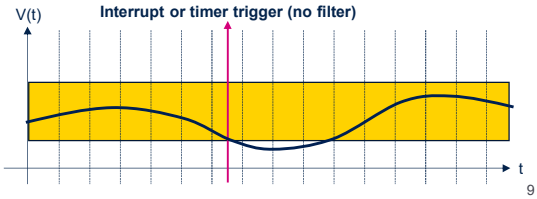
ADC provides an automatic offset calibration procedure that controls the whole calibration sequence including the ADC power-on and off.
As the offset can be dependent on the voltage or temperature, it might be necessary to re-calibrate in case of significant reference voltage or temperature change.

Analog watchdog

- The ADCs have three Window watchdogs
  - One Analog Watchdog can monitor one selected channel or all enabled channels
  - Two Analog Watchdogs can monitor several selected channels
  - Each watchdog continuously monitors an over- and/or under-threshold condition, then generates either an interrupt or an external signal
- ADC1 analog watchdog filter (only with AWD1)
  - Interrupt or signal generation only after programmable number of consecutive threshold detections



Window Watchdog	Channel to monitor
AWD1	1 (regular, injected, regular or injected) or all (regular, injected, regular and injected)
AWD2	All channels selected in ADC_AWD2CR
AWD3	All channels selected in ADC_AWD3CR



life.augmented

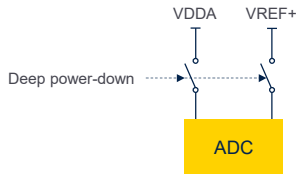
9

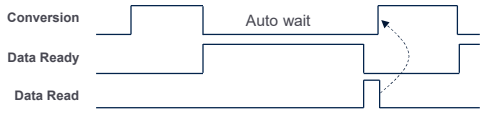
Each ADC has 3 integrated analog watchdogs with high and low threshold settings.
The ADC conversion value is compared to this window threshold, if the result exceeds the threshold, an interrupt or timer trigger signal can be asserted without CPU intervention.
Regarding ADC1, the Analog watchdog 1 also has a filtering capability.
If data is out-of-range for a number of times higher than the value specified in AWDFILT in ADCx_HTR1 register, the AWDx flag is set and the corresponding interrupt is issued.

ADC Low-power features

Several low-power features are implemented

- Deep power-down mode
  - Internal supply for ADC can be disabled by power switch for leakage reduction
- Auto-delayed conversion
  - ADC can automatically wait until last data is read
- Power consumption depends on sampling time
  - 610µA @ 5 Msamples/s
  - 170µA @ 1 Msample/s





Auto_delayed conversion

life.augmented

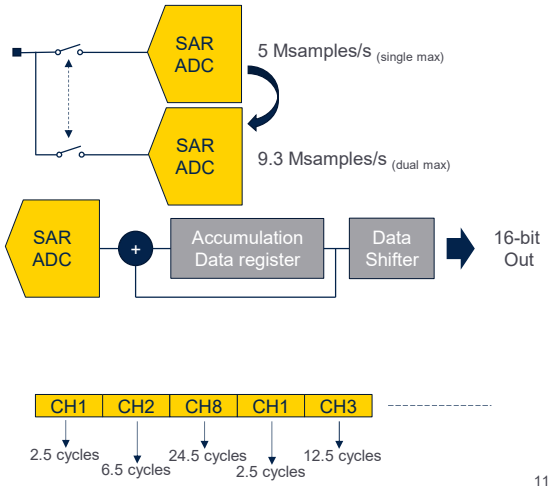
10

The STM32H5's ADCs support a Deep power-down mode. When the ADC is not used, it can be disconnected by a power switch to further reduce the leakage current.
Auto-delayed mode makes the ADC wait until the last conversion data is read before starting the next conversion.
This avoids unnecessary conversions and thus reduces power consumption.
The power consumption is in function of the sampling frequency.
For low sampling rates, the current consumption is reduced almost proportionally.

High performance features

Several high performance features are implemented

- 5 Msamples/s for 75 MHz ADC clock @12-bit conversion
- Interleave mode can support up to 9.3 Msamples/s
- Hardware oversampling
  - Accumulator and bit shifter can output 16-bit data without CPU support
- Flexible sequencer
- Auto-calibration to reduce offset



life.augmented

11

All values in this slide are related to STM32H56X and STM32H573 MCUs. For STM32H503, performance is halved.
The ADC supports up to 5 mega samples per second of 12-bit conversion at a frequency of 75 MHz.
When associating a master and a slave ADC, the interleaved mode maximum performance is one sample every 8 clocks, so 9.3 Mega sample per second at 75 MHz clock.
The ADC includes the oversampling hardware which accumulates data and then averages them without CPU assistance.
The oversampler can accommodate from 2 to 256 times samples and right shift from one to eight binary digits.

The sequencer allows the user to convert up to 16 channels in any desired order.
Also each channel can have a different sampling time.
The ADC offers an auto-calibration mechanism for the offset.
It is advisable to run calibration in the application if the reference voltage changes by more than 10%.

ADC conversion speeds

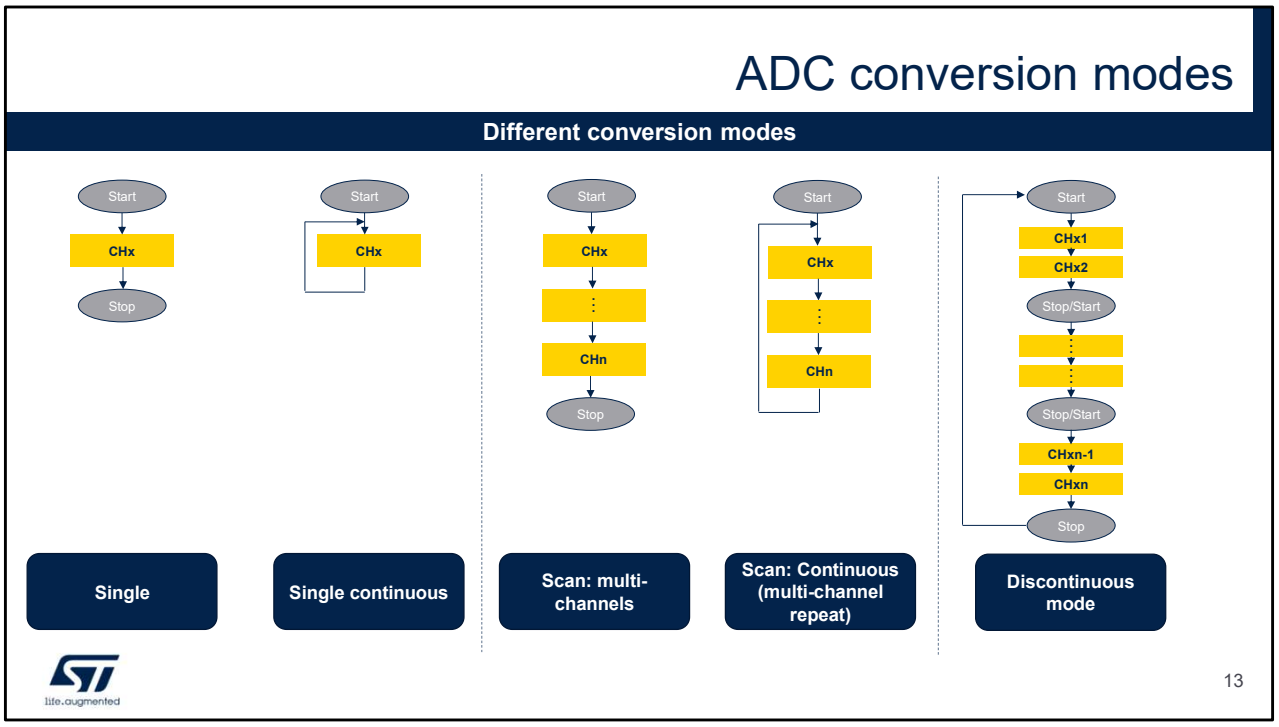
- ADC needs minimum 2.5_{ADC_CLKs} for sample period and 12.5_{ADC_CLKs} for conversion (12 bits)
- 75 MHz maximum clock with 15-cycle results in 5 Msamples/s
- Speed up by lower resolution
  - 12-bit: 12.5_{ADC_CLKs}(+2.5) ➔ 5.00 Msamples/s @75MHz
  - 10-bit: 10.5_{ADC_CLKs}(+1.5) ➔ 5.77 Msamples/s @75MHz
  - 8-bit: 8.5_{ADC_CLKs}(+1.5) ➔ 6.82 Msamples/s @75MHz
  - 6-bit: 6.5_{ADC_CLKs}(+1.5) ➔ 8.33 Msamples/s @75MHz

Resolution	$t_{Conversion}$
12 bits	15 Cycles
10 bits	13 Cycles
8 bits	11 Cycles
6 bits	9 Cycles

life.augmented

12

The global conversion time is equal to the sampling time plus the conversion time.
The ADC needs a minimum of 2.5 clock cycles for the sampling and 12.5 clock cycles for conversion for 12-bit mode.
With a 75-MHz ADC clock, it can achieve 5 mega sample per second.
For a higher sampling speed, it is possible to reduce the resolution down to 10, 8 or 6 bits.
STM32H503 maximum sampling rate is 2.5 mega sample per second for 12-bit mode.



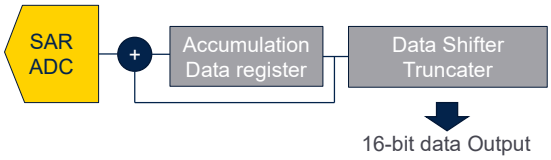
The ADC supports several conversion modes:

- Single mode, which converts only one channel, in Single-shot or Continuous mode.
- Scan mode, which converts a complete set of pre-defined programmed input channels, in Single-shot or Continuous mode.
- Discontinuous mode, which converts a short sequence (subgroup) that is part of the sequence of conversions at each trigger signal. In this figure, this short sequence includes two channels.

Hardware oversampling

Data pre-processing to offload the CPU

- Programmable oversampling ratios: x2 to x256
- Programmable data shifter & truncator right shift of 0 to 8 bits.
- Up to 16-bit data width
- Averaging, data rate reduction, SNR improvement, and basic filtering



Oversampling ratio	Output resolution	Equivalent sampling frequency (max)
x1(none)	12 bits	5 Msamples/s
x4	13 bits	1.25 Msamples/s
x16	14 bits	312.5 Ksamples/s
x256	16 bits	19.5 Ksamples/s

life.augmented

14

A built-in hardware oversampler allows improving analog performances while off-loading the related computational burden from the CPU.
It can sample by 2 to 256 times without CPU support.
The converted data is accumulated in a register and the output can be processed by the data shifter and the truncator.
12-bit data can be extended to be presented as a 16-bit data register.
This functionality can be used as an averaging function or for data rate reduction and signal-to-noise ratio improvement as well as for basic filtering.

Data transfers

Reduced software overhead

- Regular conversion data is stored in a 16-bit data register
  - Software polling, interrupts or DMA requests can be used to move the data
  - The OVERRUN flag is set when previously converted data is overwritten by current data
  - For the analog watchdogs, it is not necessary to process each data
    - The OVERRUN flag can be disabled
- Injected conversion data is stored in four 16-bit data registers
  - Injected conversion data is stored in dedicated registers
  - The regular data sequence can be kept even if injected conversion occurs

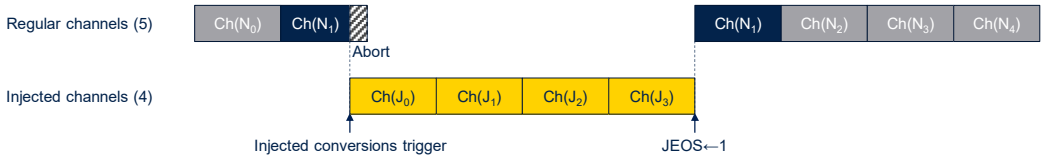
life.augmented

15

The ADC conversion result is stored in a 16-bit data register.
The system can use CPU polling, interrupts or DMA to make use of the samples obtained by the regular conversion.
An overrun flag can be generated if data is not read before the next conversion data is ready.
For injected channel conversions, 4 dedicated data registers are available, one per injected channel.
Data cannot be managed by DMA for injected channel conversions.

Injected conversions

Interruption during ADC conversion



- ADC can accept injected triggers even if a regular conversion is running
  - A trigger will stop the regular conversion then start the injected conversion
    - Up to 4 injected conversions are available by a single trigger
  - Auto-resume occurs once the injected conversion finishes
  - Four dedicated 16-bit data registers are available for the injected conversion result
  - A queue of context is implemented to anticipate up to 2 contexts for the next injected sequence of conversions
    - Queue of injected conversion can be reprogrammable on the fly

life.augmented

16

An injected conversion is used to interrupt the regular conversion, then insert up to 4 channel conversions.
Once an injected conversion is finished, the regular conversion sequence can be resumed.
The injected conversion result is stored in dedicated data registers.
Flags and interrupts are available for the end of conversion or end of sequence.
The selection of injected channels can be reprogrammed on the fly.
Even if a regular or injected conversion is in progress, you can add a different channel to the queue so that the next injected channel can be different from the previous one.

Interrupts and DMA

Interrupt event	Description	Interrupt event	Description
ADRDY	The ADC is ready to convert	AWDx	An analog watchdog threshold breach detection occurs
EOC	The end of regular conversion	EOSMP	The end of a sampling phase
EOS	The end of sequence for regular conversion group	OVR	A data overrun occurs
JEOC	The end of injected conversion	JQOVF	The injected sequence context queue overflows
JEOS	The end of sequence of an injected conversion group		

- DMA requests can be generated after each end of conversion of a channel

life.augmented

17

Each ADC supports 9 interrupt events:

- ADC ready
- End of conversion
- End of sequence
- End of injected conversion
- End of injected sequence
- Analog watchdog
- End of sampling
- Data overrun
- Overflow of the injected sequence context queue.

DMA requests can be generated at each end of conversion when the ADC output data is ready.

Low-power modes

MPU / MCU domain state	Description of the peripheral allocated to the domain
Run	**Active**
Sleep	**Active** ➢ Peripheral interrupts cause the device to exit Sleep mode
Stop	**No operation** ➢ Peripheral registers content is kept
Standby or Shutdown	**Powered-down** ➢ The peripheral must be reinitialized after exiting Standby mode

- In Deep power-down mode, the analog part of each ADC is switched off by an on-chip power switch
  - Calibration data is kept

life.augmented

The ADCs are active in Run and Sleep modes.
In Stop mode, the ADCs are not available, but the contents of their registers are kept.
In Standby mode, the ADCs are powered-down and must be reinitialized when returning to a higher power state.
There is a Deep power-down mode in each ADC itself which reduces leakage by turning off an on-chip power switch. This is the recommended mode whenever an ADC is not used.

Performance

Performance for STM32H56X/57X BGA package			
	Condition	Data (typ.)	Unit
Sampling rate	12-bit mode	5	Msamples/s
	10-bit mode	5.77	Msamples/s
	8-bit mode	6.82	Msamples/s
Differential Non Linearity (DNL)	(single ended)	±0.75	LSB
Integral Non Linearity (INL)	12-bit mode	±2	LSB
Effective Number Of Bits (ENOB	12-bit mode (single ended)	10.4	bits
	12-bit mode (differential)	11.4	bits
Consumption	5 Msamples/s	810	µA
	1 Msamples/s	244	µA

life.augmented

19

The following table shows performance parameters for the ADC regarding the STM32H562 and H57X BGA packages.
For STM32H503, please refer to the data sheet.
All values are preliminary.
Performance is package dependent, please refer to the datasheet or application note for the package and product dependent information.

Related peripherals

- Refer to training modules linked to this peripheral, if needed:
  - DMA: Direct memory access controller
  - NVIC: Nested Interrupt Vectored Interrupt Controller
  - GPIO: General-purpose inputs and outputs
  - RCC: Clock module
  - TIM: Timers for triggering interrupts and events

life.augmented

These peripherals may need to be specifically configured for correct use with the ADCs.
Please refer to the corresponding peripheral training modules for more information.

References

- For more details, please refer to the following resources:
  - Application note AN2834: How to get the best ADC accuracy in STM32Fx Series and STM32L1 Series devices
  - Application note AN4073: How to improve ADC accuracy when using STM32F2xx and STM32F4xx microcontrollers
  - Application note AN2668: Improving STM32F1x and STM32L1x ADC resolution by oversampling
  - Application note AN4629: ADC hardware oversampling for microcontrollers of the STM32 L0 and L4 series

life.augmented

Several application notes dedicated to analog-to-digital converters are available.
To learn more about ADCs, you can visit a wide range of web pages discussing successive approximation analog-to-digital converters.

Digital to analog converter (DAC)

Now let’s look at the features of the DAC.

DAC

- One DAC interface, two outputs
- When the DAC output is connected to the OPAMP/Comp/ADC, the GPIO can be used for other functionalities
- Sample and hold mode for stop mode
  - It reduce the current static current consumption when output need to be kept during the stop mode
- The DAC kernel clock can be common to the ADCs to avoid perturbation

life.augmented

23

The STM32H5 integrates one DAC module with two analog outputs.
The DAC output can be routed to internal resources, such as a comparator and operational amplifier, without using GPIO.
In stop mode, the DAC can support the Sample and Hold mode, using the LSI/LSE clock source for static conversion.
To synchronize DAC and ADC, the same clock source can be used for both peripherals.
This is done by selecting the dac_ker_ck clock instead of the dac_hclk clock (AHB clock) in the RCC.

DAC specification

		Typical	Unit
VDDA		1.8 ~ 3.6	V
Resolution		12	bits
Differential Non Linearity (DNL)		+/- 2	LSB
Integral Non Linearity (INL)		+/- 4	LSB
Consumption	Buffer On	660	µA
	Buffer Off	180	µA
Settling time	Buffer On (50pF)	2.05	µsec
	Buffer Off (10pF)	1.7	µsec

life.augmented

24

This table shows the typical characteristics of the STM32H5's DAC.
The STM32H5's DAC can work between 1.8 and 3.6 volts whether the buffer state is on or off.
10-bit monotonicity is guaranteed.
By using Sample & Hold mode, the current consumption can be drastically reduced.
The DAC can handle a sampling rate of 1 mega sample per second.

Thank you

life.augmented

Several application notes dedicated to analog-to-digital converters are available.
To learn more about ADCs, you can visit a wide range of web pages discussing successive approximation analog-to-digital converters.
In addition to this presentation, you can refer to the following presentations:

- Reset and Clock Control
- Power management.