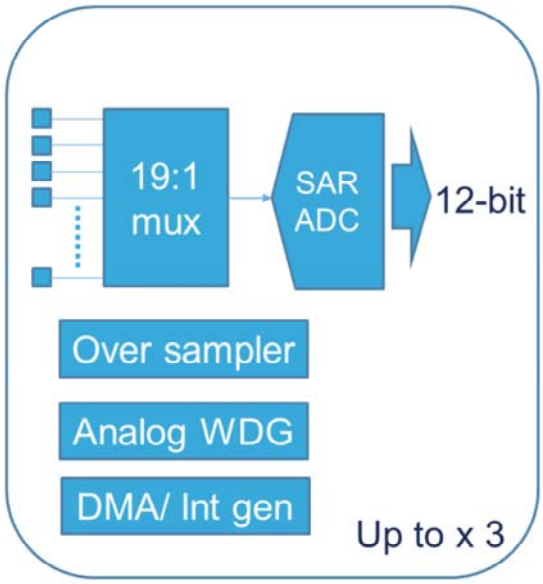
Hello and welcome to this presentation of the STM32L4 Analog-to-Digital Converter block. It will cover the main features of this block, which is used to convert the external analog voltage-like sensor outputs to digital values for further processing in the digital domain.

Overview



- Provides analog-to-digital conversion

	L47x/48x/49x/4Ax	L43x/44x/45x/46x	L41x/42x
ADC	x3	x1	x2
Input Channels	16-24	10-16	10-16

- 12-bit resolution, 16-bit with oversampling
- 5.33 Msamples/s max. (12-bit)
- Three analog watchdogs per ADC
- DMA request generation
- Interrupt generation

Application benefits

- Ultra-low power consumption: 210 µA @ 1 Msample/s
- Flexible trigger, data management to offload CPU

life.augmented

The analog-to-digital converters inside STM32 products allow the microcontroller to accept an analog value like a sensor output and convert the signal into the digital domain. There are up to three ADCs and 16 to 24 analog inputs depending on the device. The ADC module itself is a 12-bit successive approximation converter with additional oversampling hardware. Under certain conditions, the oversampled output can have a 16-bit result. The sampling speed is more than five mega samples per second. Each ADC module integrates three analog watchdogs. The data can be made available either through DMA movement or interrupts. This ADC is designed for low power and high performance. There are a number of triggering mechanisms and the data management can be configured to minimize the CPU workload.

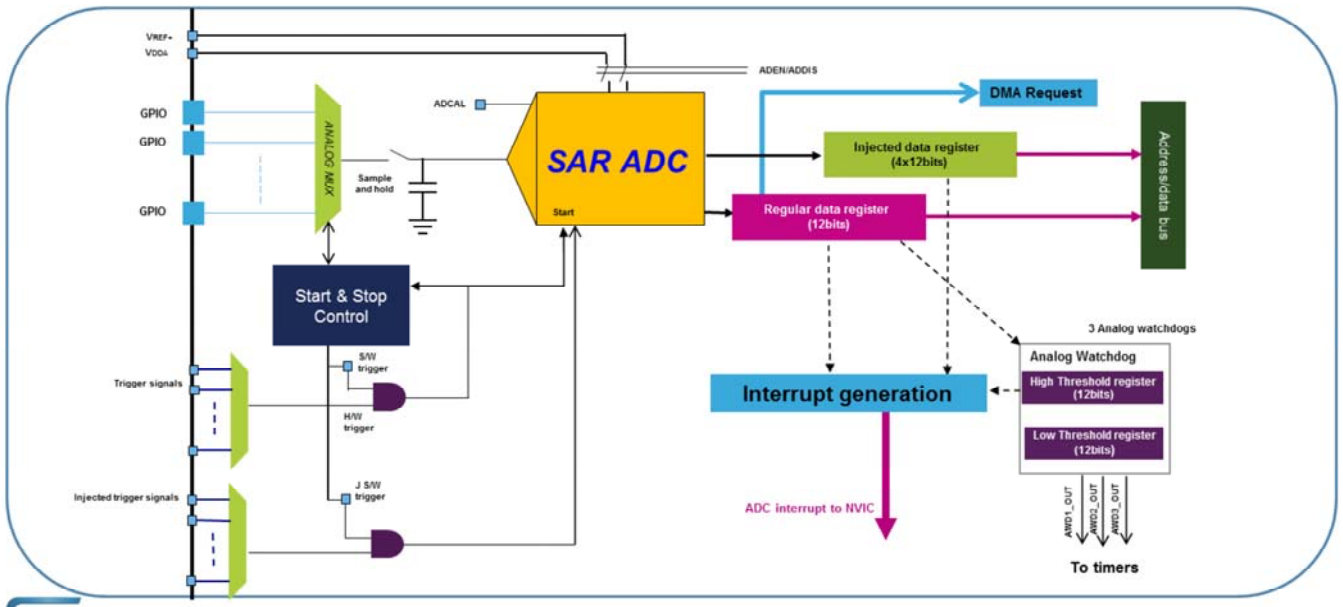
Key features

ADC units	Up to 3 modules
Input channel	Up to 24 external channels (GPIOs), single/differential
Technology	12-bit successive approximation
Conversion time	188 nS, 5.33 Msamples/s (when f_{ADC_CLK} = 80 MHz)
Functional mode	Single, Continuous, Scan, Discontinuous, or Injected
Triggers	Software or external trigger (for Timers & IOs)
Special functions	Hardware oversampling, analog watchdogs
Data processing	Interrupt generation, DMA requests
Low-power modes	Deep power-down, auto delay, power consumption dependent on speed

life.augmented

Up to 3 analog-to-digital converters are integrated inside STM32L4x6 products. The input channel is connected to up to 24 GPIO channels capable of converting signals in either Single-end or Differential mode. The ADCs can convert signals in excess of 5 mega samples per second. There are several functional modes which will be explained later. There are also several different triggering methods. In order to offload the CPU, the ADC has 3 analog watchdogs for monitoring thresholds. The ADC also offers oversampling to extend the number of bits presented in the final conversion value. For power-sensitive applications, the ADC offers a number of low-power features.

Block diagram



life.augmented

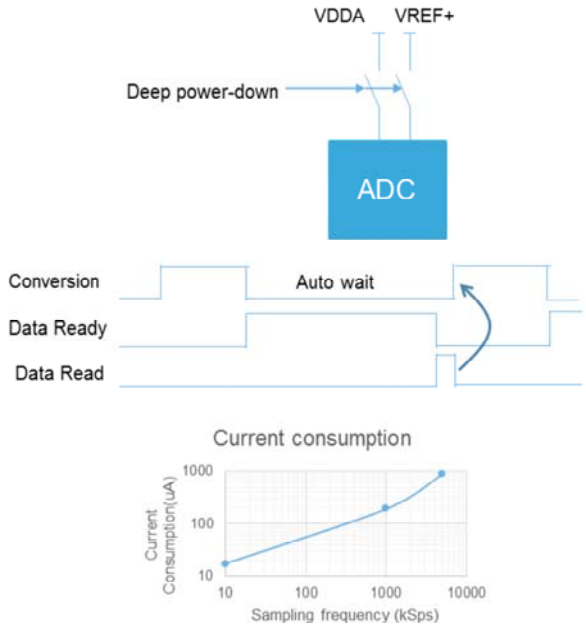
Up to x3

This slide shows the general block diagram for each analog-to-digital converters embedded in the STM32L4.

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
  - 865 µA @ 5 Msamples/s, 190 µA @ 1 Msample/s, 17 µA @ 10 ksamples/s

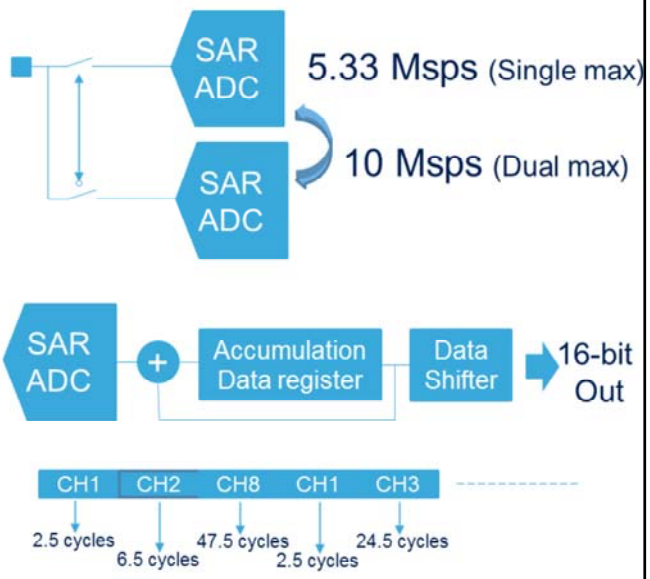


The STM32L4's ADCs support a Deep power-down mode. When the ADC is not used, it can be disconnected by a power switch to further reduce the leakage current. Auto-delayed mode makes the ADC wait until the last conversion data is read before starting the next conversion. This avoids unnecessary conversions 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

- 5.33 Msamples/s for 80 MHz ADC clock
- Interleave mode can support up to 10 Msamples/s (devices with dual ADC)
- Hardware oversampling
  - Accumulator and bit shifter can output 16-bit data without CPU support
- Flexible sequencer
- Auto-calibration to reduce offset



life.augmented

The ADC supports up to 5.33 mega samples per second of conversion. By using dual interleaved mode, it can be extended to ten mega samples par second. The ADC includes the oversampling hardware which accumulates data and then divides without CPU help. 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 different sampling period. The ADC offers an auto calibration mechanism. It is recommended to run the calibration on the application if the reference voltage changes more than 10% so this would include emerging from RESET or from a low power state where the analog voltage supply has been removed and reinstated.

ADC conversion speeds

Conversion speed is resolution dependent

- ADC needs minimum 2.5_{ADC_CLKs} for sample period and 12.5_{ADC_CLKs} for conversion (12-bit).
- 80 MHz maximum clock with a 15 cycle results in 5.33 Msamples/s
- Speed up by low resolution
  - 10-bit : 10.5_{ADC_CLKs}(+2.5) => 6.15 Msamples/s
  - 8-bit : 8.5_{ADC_CLKs} (+2.5) => 7.27 Msamples/s
  - 6-bit : 6.5_{ADC_CLKs} (+2.5) => 8.88 Msamples/s

Resolution	$t_{Conversion}$
12 bits	12.5 Cycles
10 bits	10.5 Cycles
8 bits	8.5 Cycles
6 bits	6.5 Cycles

life.augmented

The ADC needs a minimum of 2.5 clock cycles for the sampling and 12.5 clock cycles for conversion. With an 80 MHz ADC clock, it can achieve 5.33 mega samples per second. For higher speed sampling, it is possible to reduce the resolution down to 6 bits then the sampling speed can go up to 8.88 mega samples per second.

Sampling times

Programmable sampling time

- The following sampling times can be selected:
  - 2.5 cycles
  - 6.5 cycles
  - 12.5 cycles
  - 24.5 cycles
  - 47.5 cycles
  - 92.5 cycles
  - 247.5 cycles
  - 640.5 cycles
- If Scan mode is selected, each input channel can have a different sampling time
  - One ADC can scan the different input source independent of the source impedance.

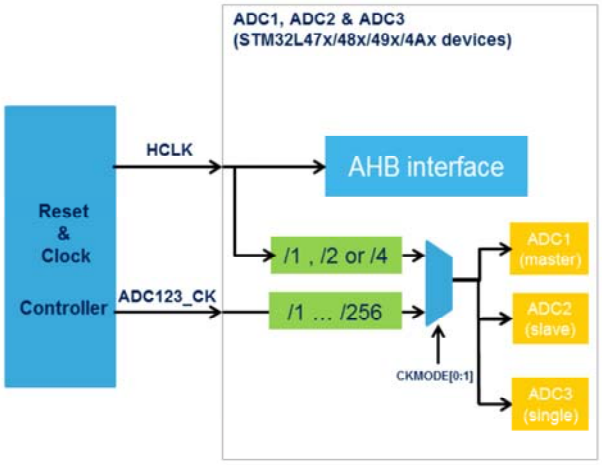
life.augmented

The sampling time can be programmed individually for each input channel of the analog-to-digital converters. The sampling times listed in this slide in ADC clock cycles are available. Longer sample times ensure that signals having a higher impedance are correctly converted.

ADC clocks

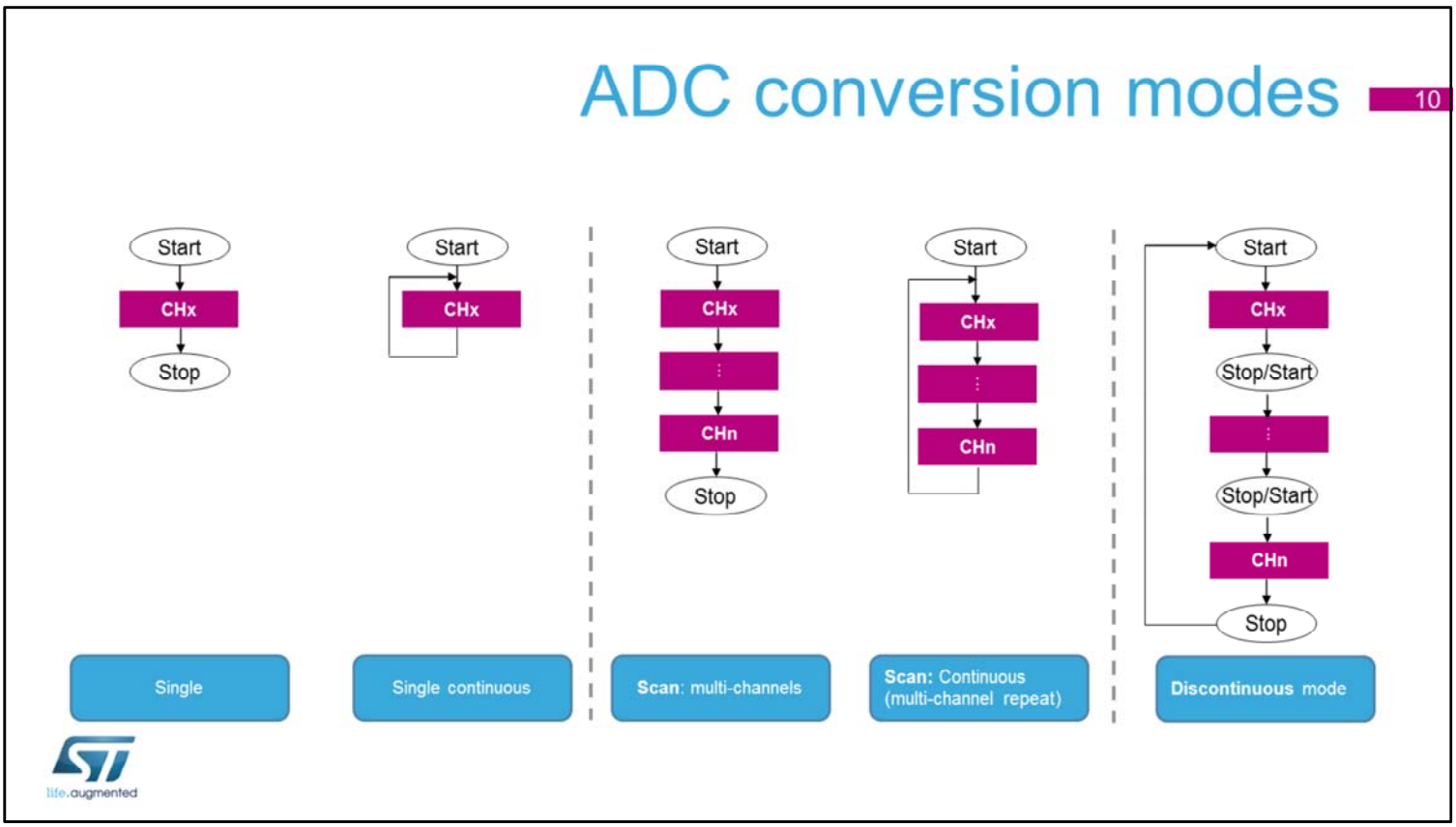
Flexible clock selection

- ADC clock can be selected from
  - AHB clock divided by 1, 2 or 4.
    If a trigger event depends on the AHB clock, the latency between event and start of conversion is deterministic.
  - Dedicated ADC clock(1)
    Independent and asynchronous to the system clock (AHB). The CPU can run slowly even if the ADC is running full-speed.



(1) Not supported by STM32L41x/42x devices

The ADCs have a selectable clock source. When the system needs to run synchronously, the AHB clock source is the best selection. If a slow CPU speed is required, but the ADC needs a higher sampling rate, the dedicated ADC clock can be selected.



AD converter 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, converts only a single channel at each trigger signal from the list of pre-defined programmed input channels.

Hardware oversampling

11

Data pre-processing to offload the CPU

- Programmable oversampling ratios: x2, x4, x8, x16, x32, x64, x128, x256
- Programmable data shifter & truncater Right shift of 0 to 8 bits
- Up to 16-bit ADC results
- Averaging, data rate reduction, SNR improvement, and basic filtering

Oversampling ratio	Output resolution	Equivalent sampling frequency max
x4	13 bits	1.33 Msamples/s
x16	14 bits	333 ksamples/s
x64	15 bits	83 ksamples/s
x256	16 bits	20.8 ksamples/s

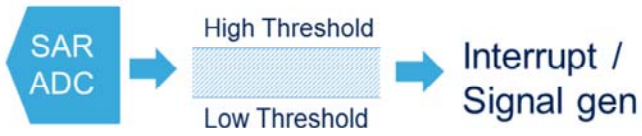
life.augmented

The ADCs support hardware oversampling. They can sample by 2, 4, 8, 16, 32, 64, 128 and 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 truncater.
12-bit data can be extended to be presented as 16-bit data. 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.

Analog watchdog

Reduced software overhead

- Each ADC has three Window comparators
  - One 12-bit analog watchdog can monitor one selected channel or all enabled channels
  - Two 8-bit analog watchdogs can monitor several selected channels
- Each watchdog continuously monitors an over- and/or under-threshold condition, then generates either an interrupt or external signal or stops a timer.



life.augmented

Each ADC has three 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 external signal can be generated or a timer can be immediately stopped without CPU intervention.

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 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

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 conversion data. 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.

Injected conversions

Interruption during of the 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 triggers.
  - Auto-resume occurs once the injected conversion finishes.
  - Four dedicated 16-bit data registers are available for the injected conversion result.
  - Creates the Interrupt, or flags for use by the user's firmware.
  - Queue of injected conversion can be reprogrammable on the fly.

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 choices for an injected channel 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 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 conversion of a channel.

life.augmented

Each ADC can generate 9 different interrupts: ADC Ready, end of conversion, end of sequence, end of injected conversion, end of injected sequence, analog watch dog, end of sampling, data overrun and the 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

Mode	Description
Run	Active.
Sleep	Active. Peripheral interrupts cause the device to exit Sleep mode.
Low-power run	Active.
Low-power sleep	Active. Peripheral interrupts cause the device to exit Low-power sleep mode.
Stop 0/Stop 1	Not available. Peripheral registers content is kept.
Stop 2	Not available. Peripheral registers content is kept.
Standby	Powered-down. The peripheral must be reinitialized after exiting Standby mode.
Shutdown	Powered-down. The peripheral must be reinitialized after exiting Shutdown 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.

The ADCs are active in Run, Sleep, Low-power run and Low-power sleep modes. In Stop0, Stop 1 or Stop 2 mode, the ADCs are not available but the contents of their registers are kept. In Standby or Shutdown 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

	Condition	Data (typ)	Unit
Sampling rate	12-bit mode	5.33	Msamples/s
	6-bit mode	8.89	Msamples/s
DNL		+/-1	LSB
INL		+/-1.5	LSB
ENOB	Single End	10.3	bits
	Differential	10.9	bits
Consumption	5 Msamples/s	865	µA
	1 Msample/s	190	µA
	10 ksamples/s	17	µA

life.augmented

The following table shows performance parameters for the ADC.

Related peripherals

- Refer to these peripherals trainings linked to this peripheral, if any
  - DMA – Direct memory access controller
  - Interrupts
  - GPIO – General-purpose inputs and outputs
  - RCC – Clock module
  - DAC – Digital-to-analog converter
  - 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.

Features for each individual ADC

ADC features	ADC1	ADC2 (1)	ADC3 (2)
Dual mode	Master	Slave	-
Internal channel connection	Bandgap Temp sensor VBAT DAC1 Out (3) DAC2 Out (3)	DAC1 Out DAC2 Out	DAC1 Out DAC2 Out Temp sensor VBAT

(1) only on STM32L47x/86x, STM32L49x/4Ax and STM32L41x/42x devices
(2) only on STM32L47x/86x and STM32L49x/4Ax devices
(3) only on STM32L43x/44x and STM32L45x/46x devices

life.augmented

The STM32L4 embeds up to three ADCs. ADC 1 and ADC 2 can be configured to work together in Dual mode, so that each analog-to-digital conversion can be synchronized between the two modules. ADC 3 works as a standalone converter.

References

- For more details, please refer to following sources
  - AN2834: How to get the best ADC accuracy in STM32 microcontrollers
  - AN4073: How to improve ADC accuracy when using STM32F2xx and STM32F4xx microcontrollers
  - AN2668: Improving STM32F1 Series, STM32F3 Series and STM32Lx Series ADC resolution by oversampling

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.