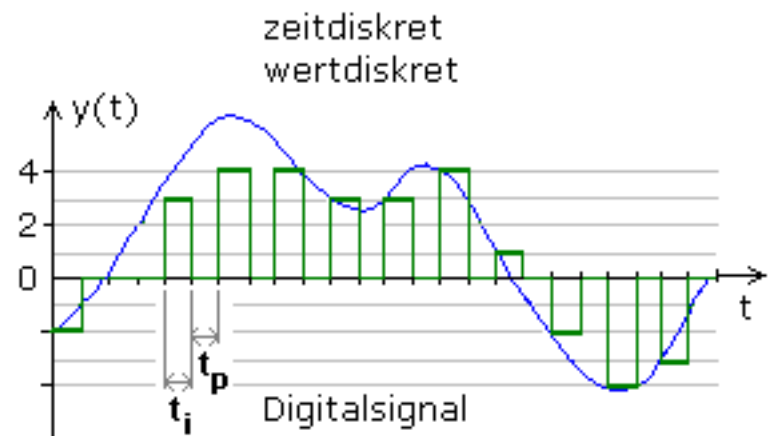
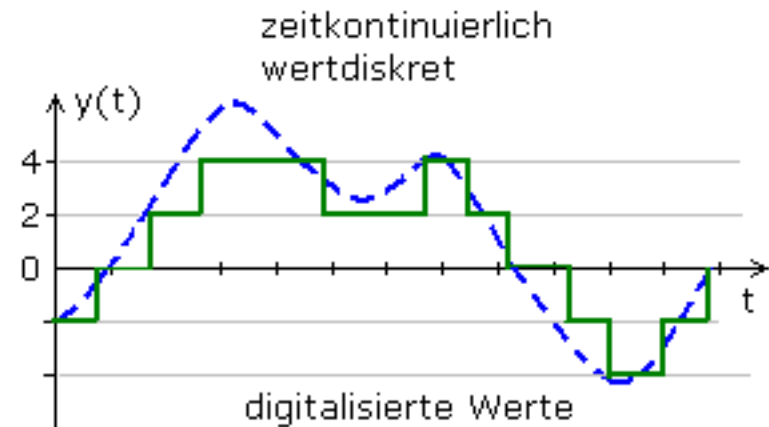
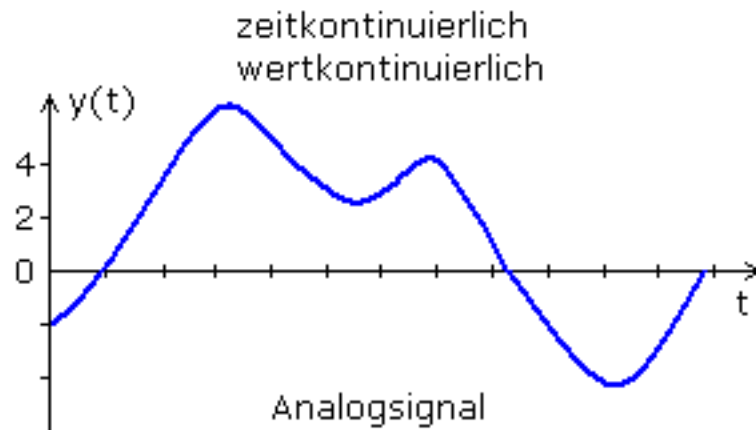


# Vorlesungsskript zu „Vertiefung Programmieren“ Input Processing

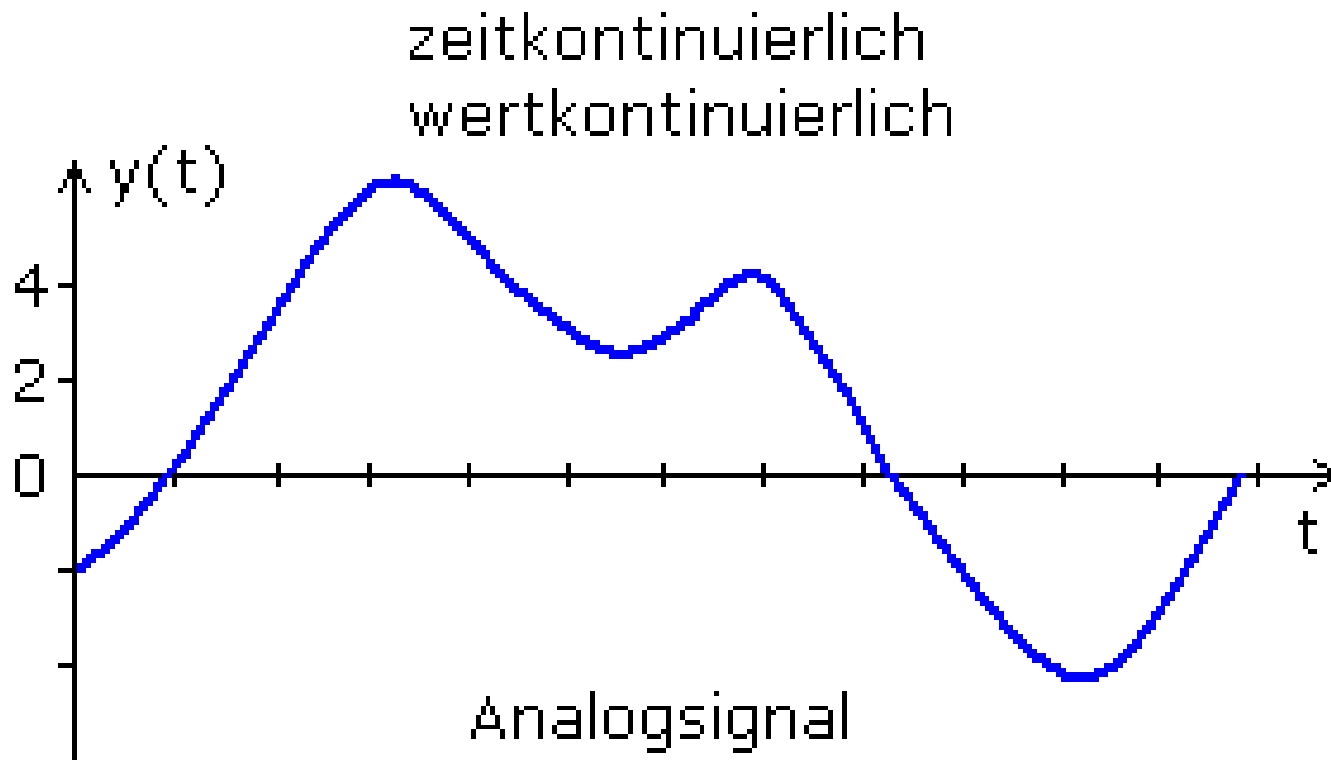


**Dozent:** Dipl.-Inf. (FH) Andreas Schmidt

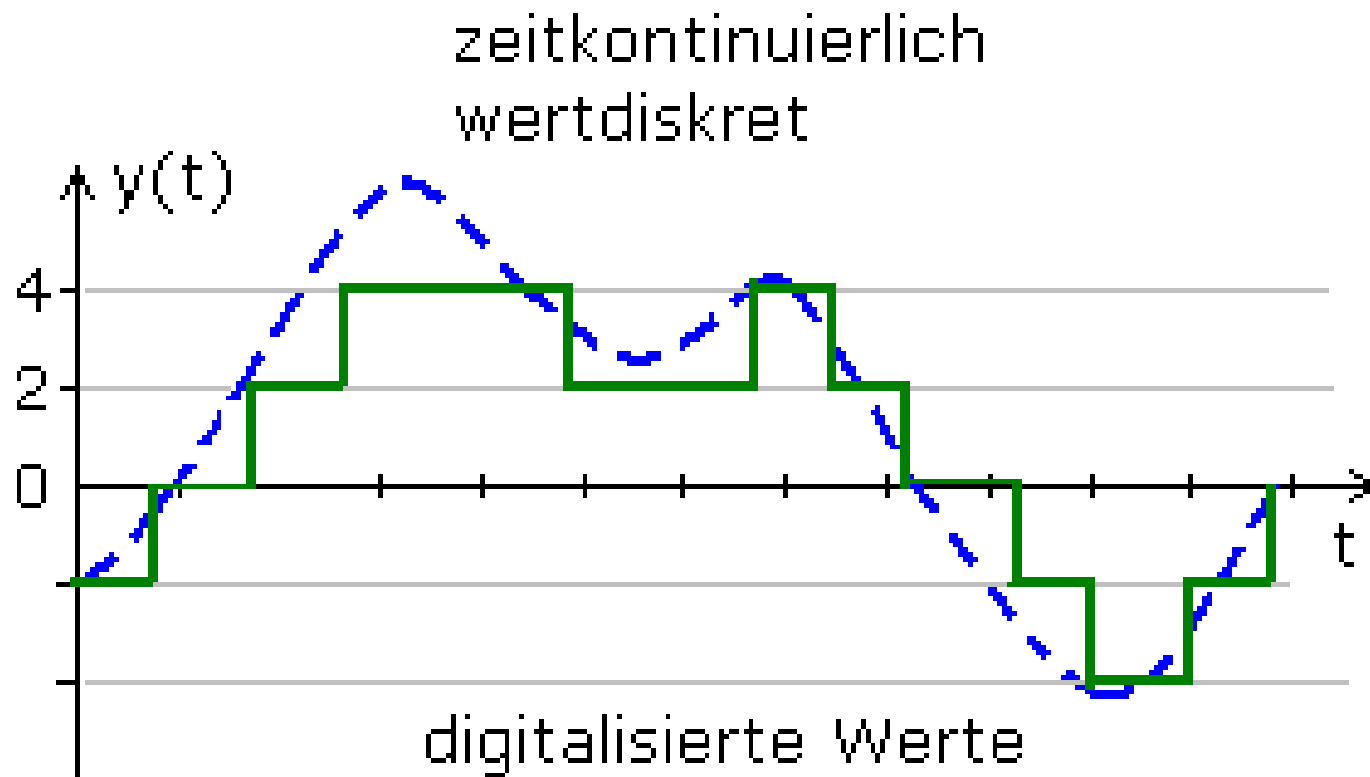
## Verlaufsarten von Signalen



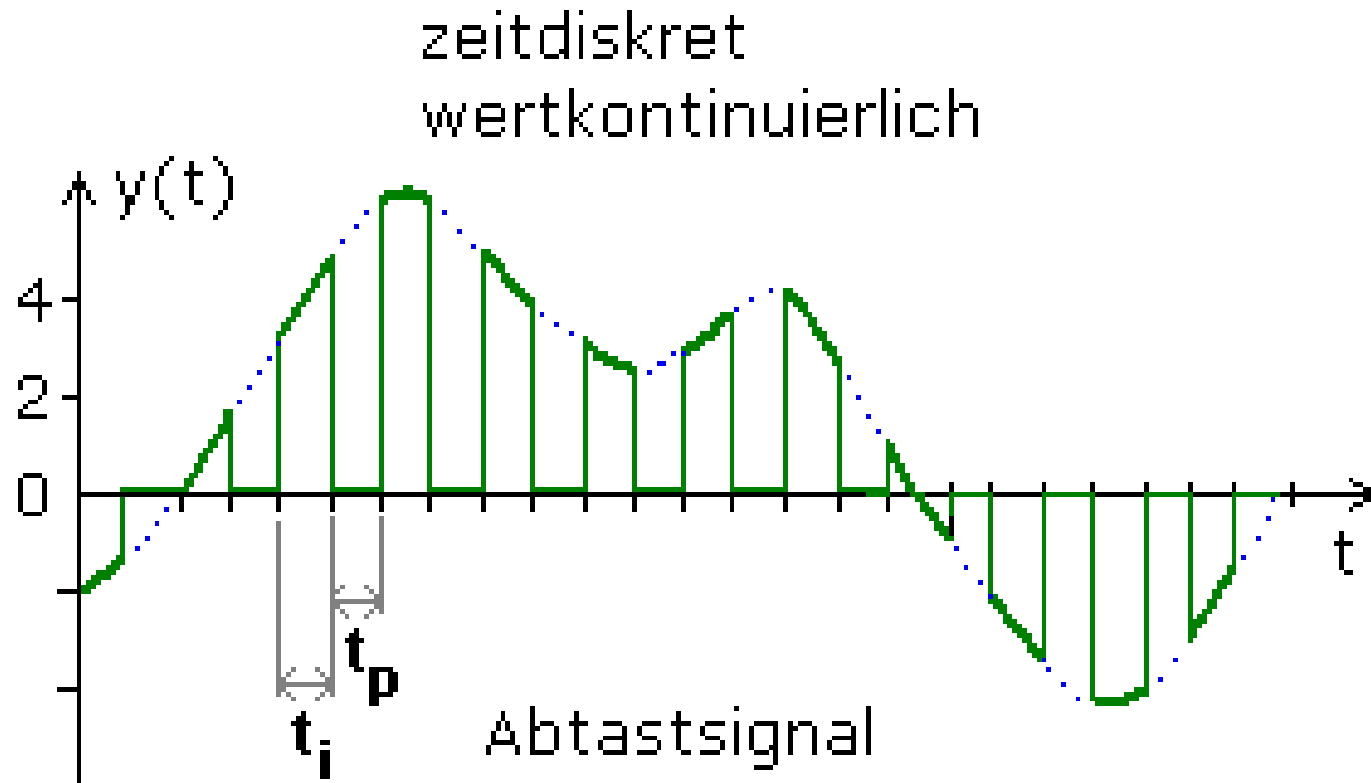
## Verlaufsarten von Signalen – Analogsignal



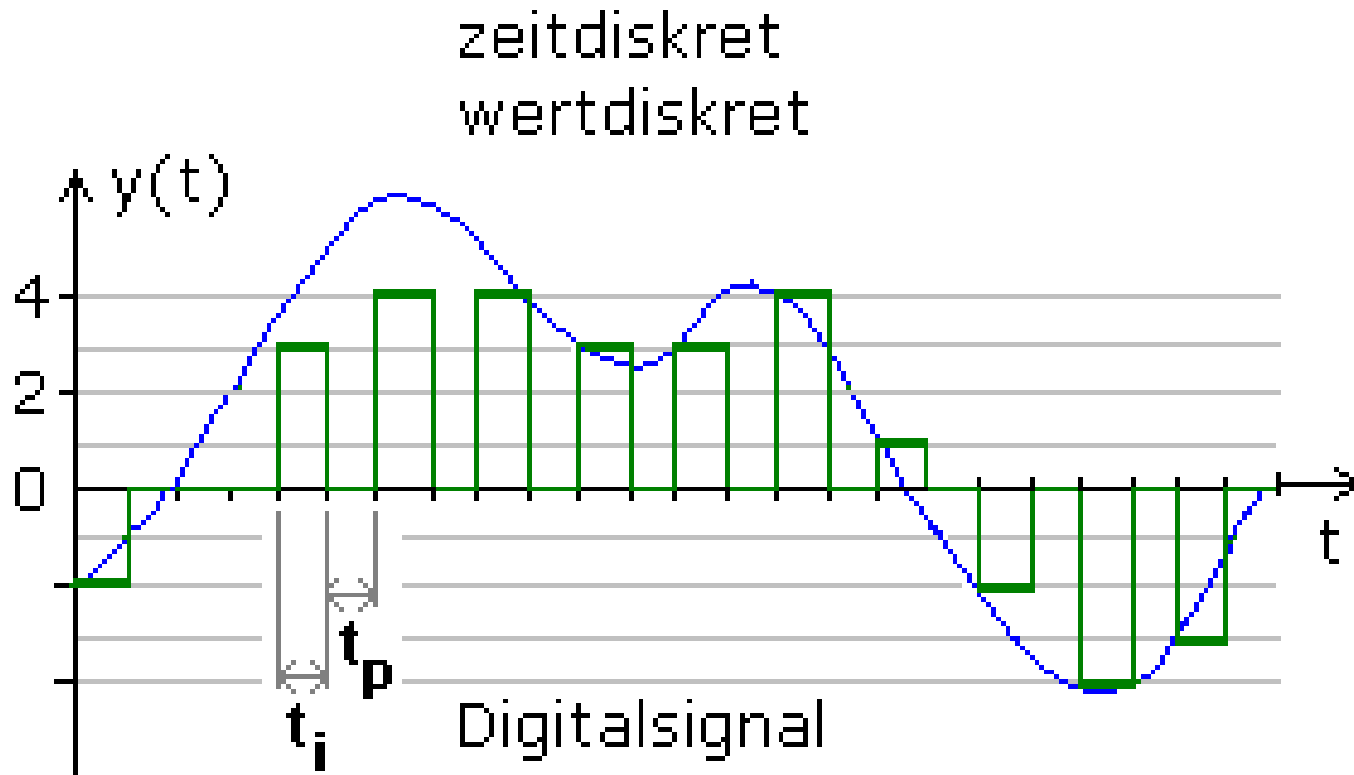
## Verlaufsarten von Signalen – Digitalisierte Werte



## Verlaufsarten von Signalen – Abtastsignal



## Verlaufsarten von Signalen – Digitalsignal



## AD-Wandler

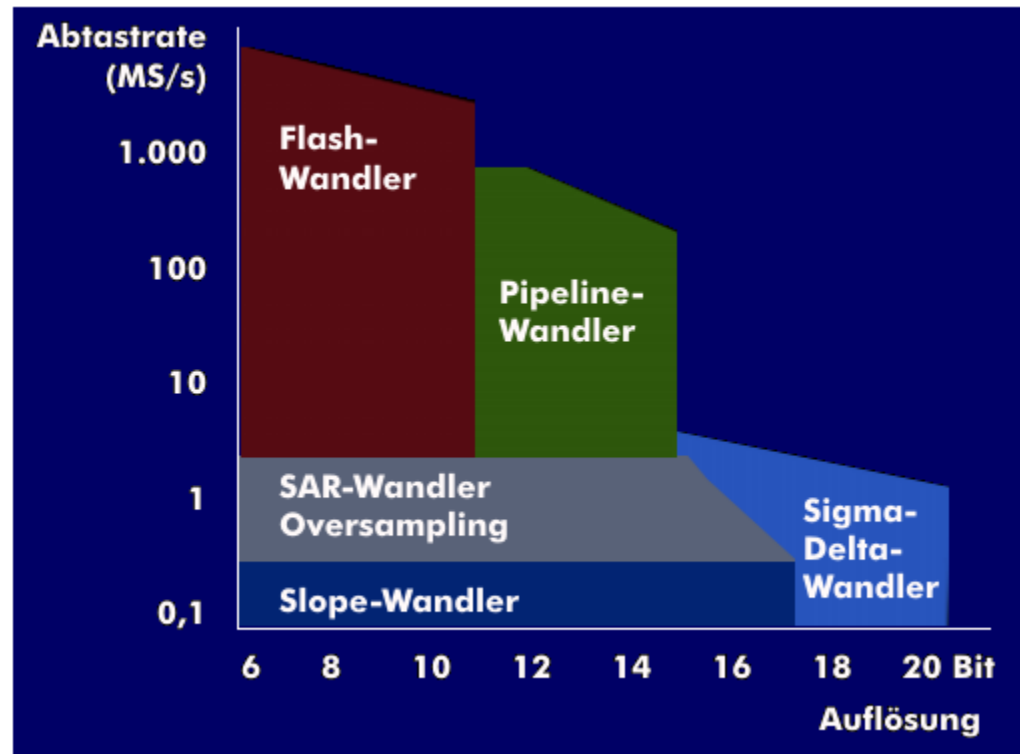
Der AD-Wandler hat die Aufgabe jeder analogen Eingangsgröße eine digitale Information in Form eines Bitmusters zuzuordnen. Die Anzahl möglicher Kombinationen eines Digitalcodes hängt von der Anzahl der zur Verfügung stehenden Bits ab

$$\text{Anzahl Werte} = 2^{\text{Anzahl Bits}}$$

z.B. ergeben sich für 8-Bits  $2^8$  Kombinationen = 256

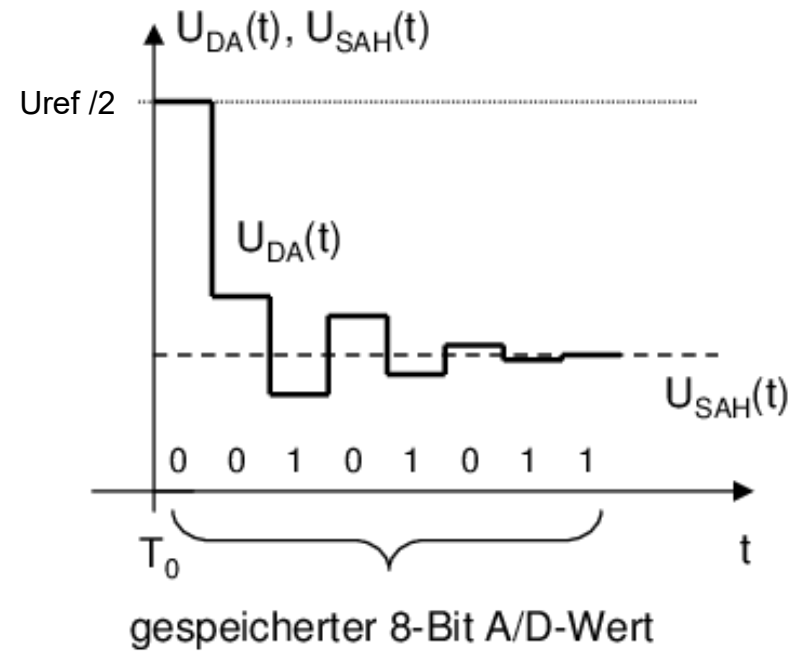
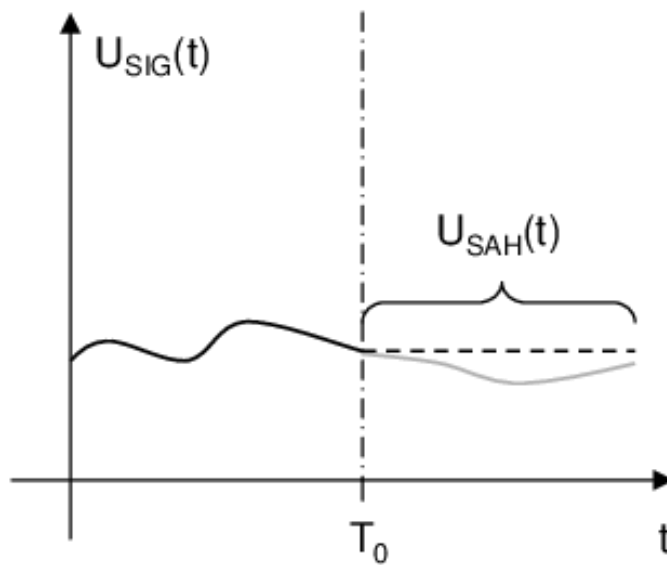
bzw.  $2^8 - 1 = 255$  Stufen

## AD-Wandler Arten



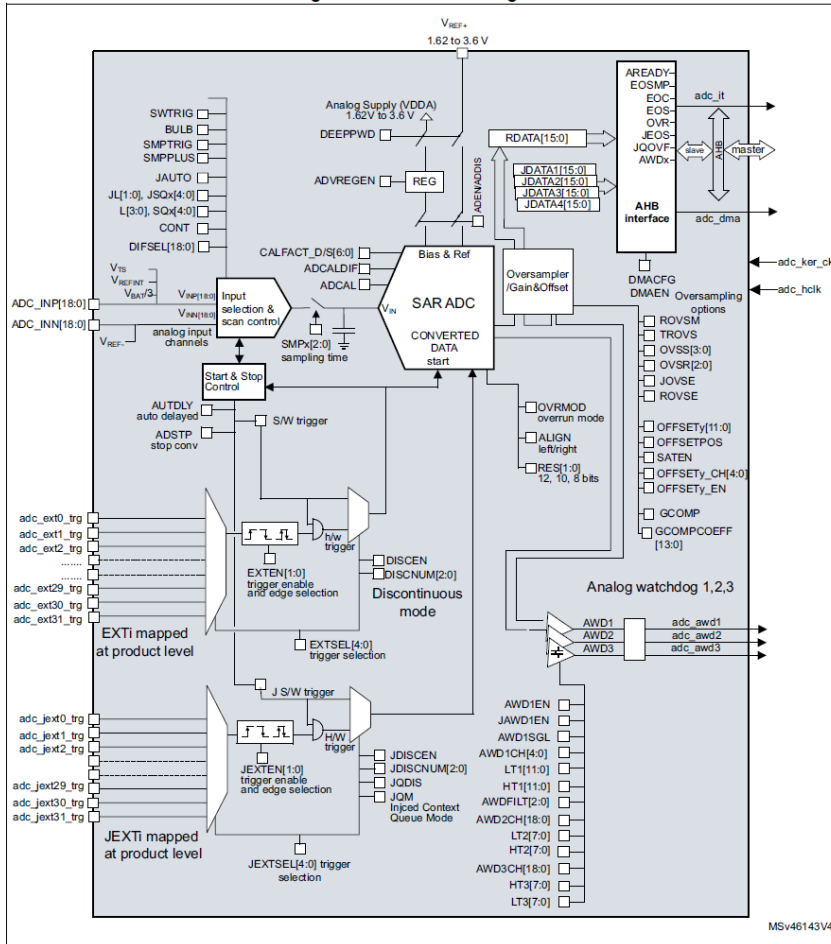


## Wägeverfahren (sukzessive Approximation)



## AD Wandler beim STM32G4

**Figure 82. ADC block diagram**



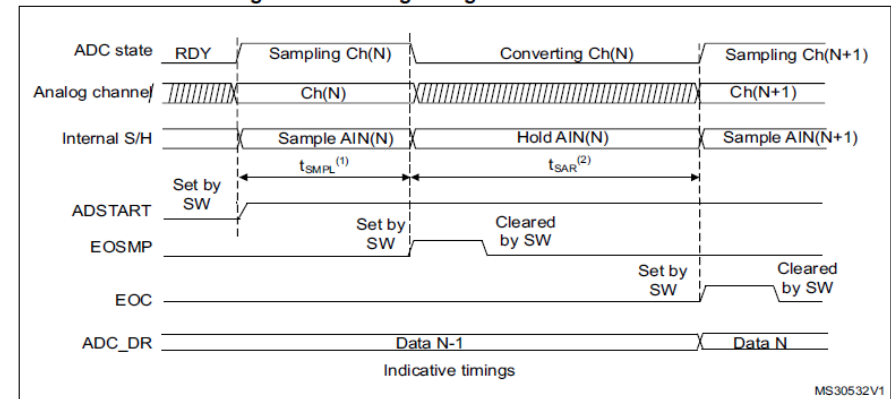
## ADC timing

The elapsed time between the start of a conversion and the end of conversion is the sum of the configured sampling time plus the successive approximation time depending on data resolution:

$$T_{\text{CONV}} = T_{\text{SMPL}} + T_{\text{SAR}} = [2.5_{\text{min}} + 12.5_{\text{12bit}}] \times T_{\text{ADC\_CLK}}$$

$$T_{\text{CONV}} = T_{\text{SMPL}} + T_{\text{SAR}} = 83.33 \text{ ns}_{|\text{min}} + 416.67 \text{ ns}_{|\text{12bit}} = 500.0 \text{ ns (for } F_{\text{ADC CLK}} = 30 \text{ MHz)}$$

**Figure 94. Analog to digital conversion time**



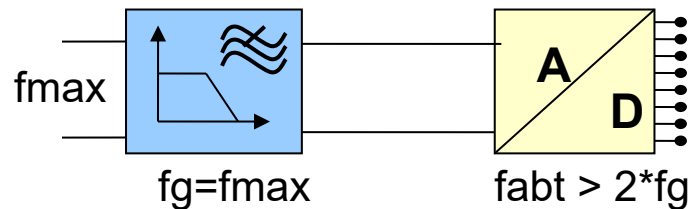
1.  $T_{SMPL}$  depends on SMP[2:0].
2.  $T_{SAR}$  depends on RES[2:0].

## Abtastung analoger Signale

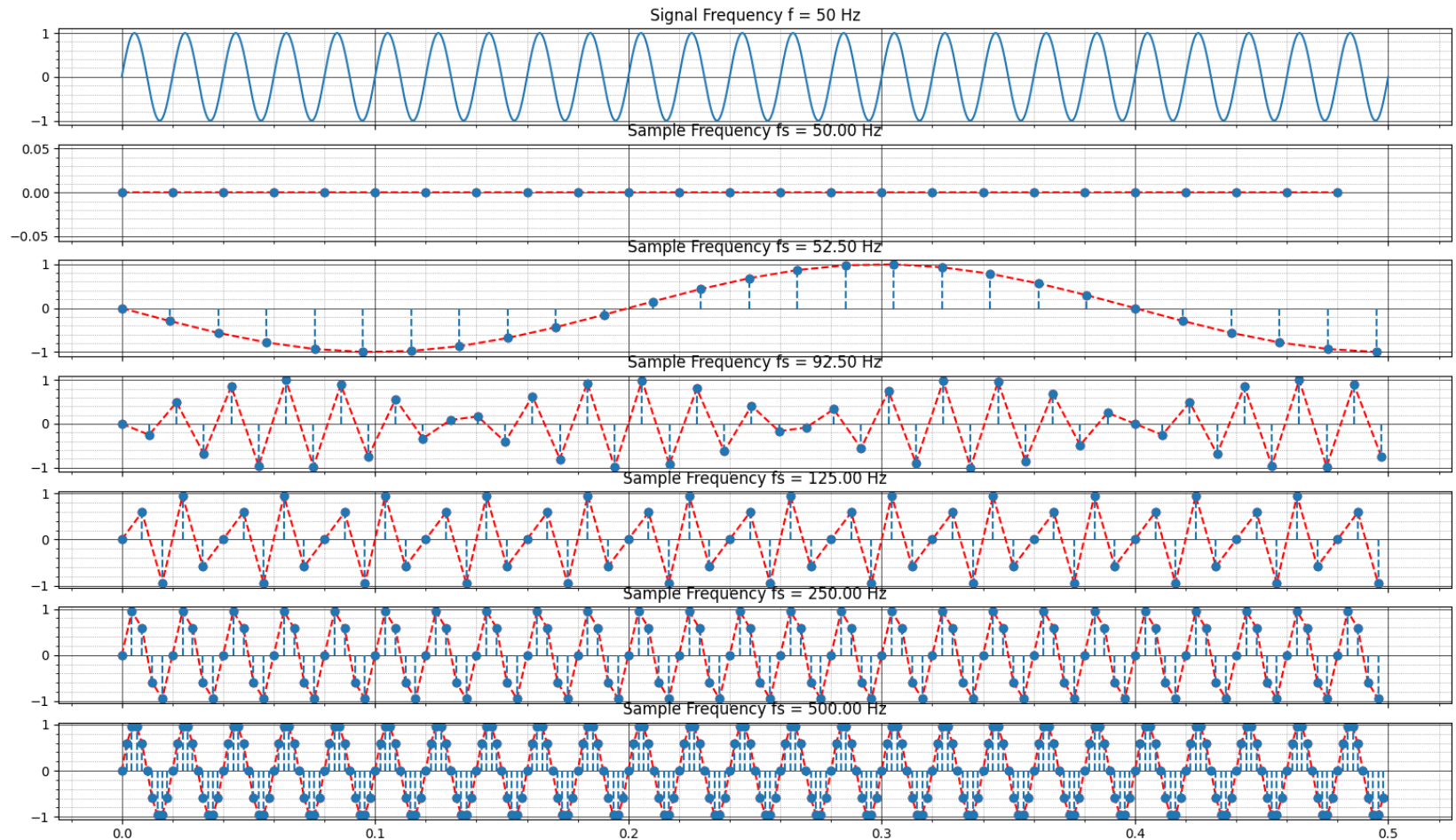
Die Abtastfrequenz muss **größer** als die doppelte Signalfrequenz sein (Shannon Theorem)

In der Praxis wird meist das 5-10fache max. Signalfrequenz verwendet, um Amplitudentreue zu erreichen

Um dies sicherzustellen, muss ein Tiefpassfilter vorgeschaltet werden (Anti Aliasing Filter)

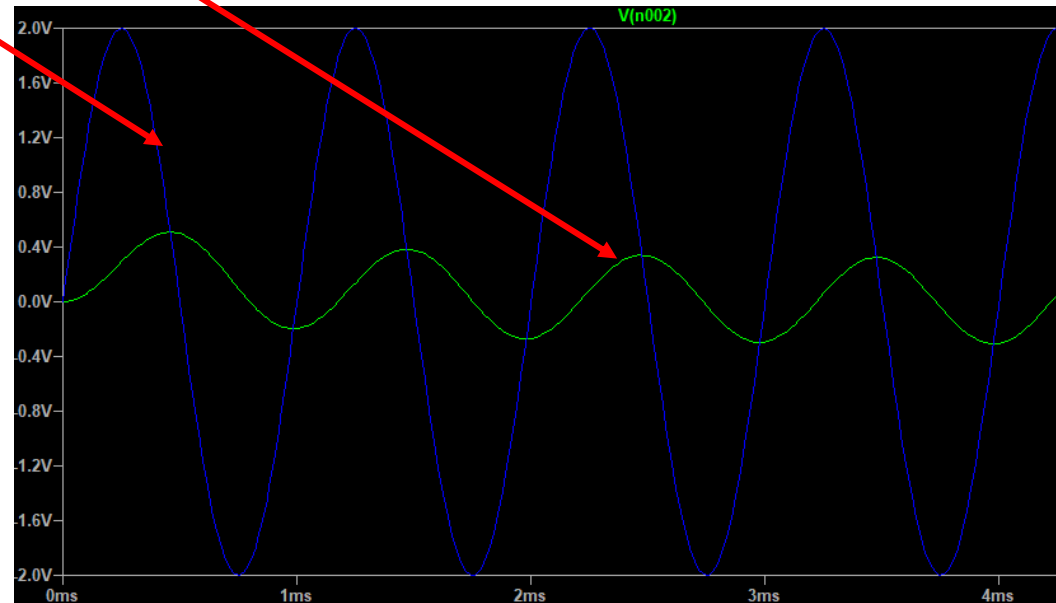
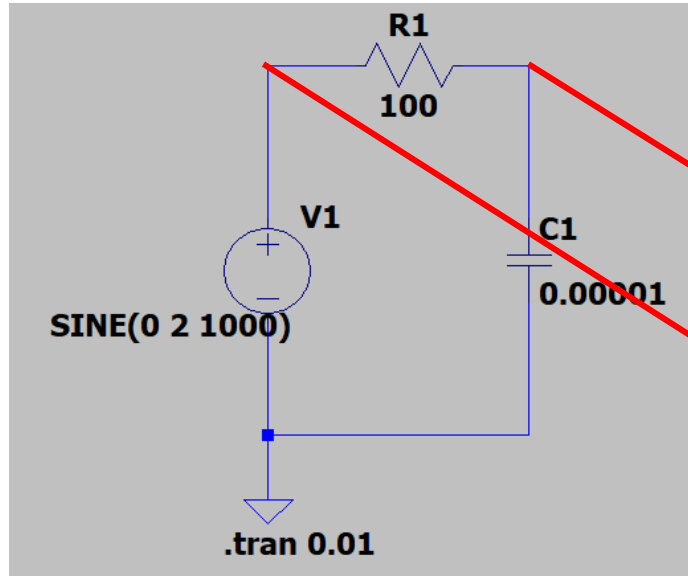


## Abtastung analoger Signale



Quelle: intern

## Anti-Aliasing Filter



## Anti-Aliasing Filter

Tiefpass zur Verhinderung von Aliasing [ [Bearbeiten](#) | [Quelltext bearbeiten](#) ]

→ Hauptartikel: *Alias-Effekt*

Wird die Abtastfrequenz zu klein gewählt, treten im digitalisierten Signal Mehrdeutigkeiten auf. Diese nichtlinearen Verzerrungen sind auch unter dem Begriff Alias-Effekt bekannt. Bei Bildern treten eventuell phasenverschobene Schatten oder neue Strukturen auf, die im Original nicht enthalten sind.

Den unteren Grenzwert der Abtastfrequenz für ein analoges Signal der Bandbreite  $f_0$

$$f_{\text{abast}} = 2 \cdot f_0$$

nennt man auch **Nyquist-Rate**. Die höchste zu übertragende Frequenz muss demnach kleiner sein als die halbe Abtastfrequenz, sonst entstehen Aliasingfehler. Aus diesem Grund werden höhere Frequenzen aus dem analogen Signal mit einem Tiefpass herausgefiltert. Die Aliasingfehler sind **Alias-Signale** (**Störsignale**, Pseudosignale), die sich bei der Rekonstruktion als störende Frequenzanteile bemerkbar machen. Wird zum Beispiel ein Sinussignal, das eine Frequenz von 1600 Hz hat, mit einer Abtastfrequenz von 2000 Hz digitalisiert, erhält man ein 400-Hz-Alias-Signal (2000–1600 Hz). Bei einer Abtastfrequenz über 3200 Hz entsteht dagegen kein Alias-Signal. Eine Abtastfrequenz von bspw. 3300 Hz führt zu einem Differenzsignal von 1700 Hz (3300–1600 Hz). Dieses ist jedoch größer als die halbe Abtastrate und wird demnach bei der Rekonstruktion durch einen Tiefpass entfernt.

In der Praxis gibt es keinen **idealen Tiefpass**. Er hat immer einen gewissen Übergangsbereich zwischen praktisch keiner Dämpfung im Durchlassbereich und praktisch vollständiger Dämpfung im Sperrbereich. Daher verwendet man in der Praxis eine modifizierte Formel zur Bestimmung der Abtastfrequenz:

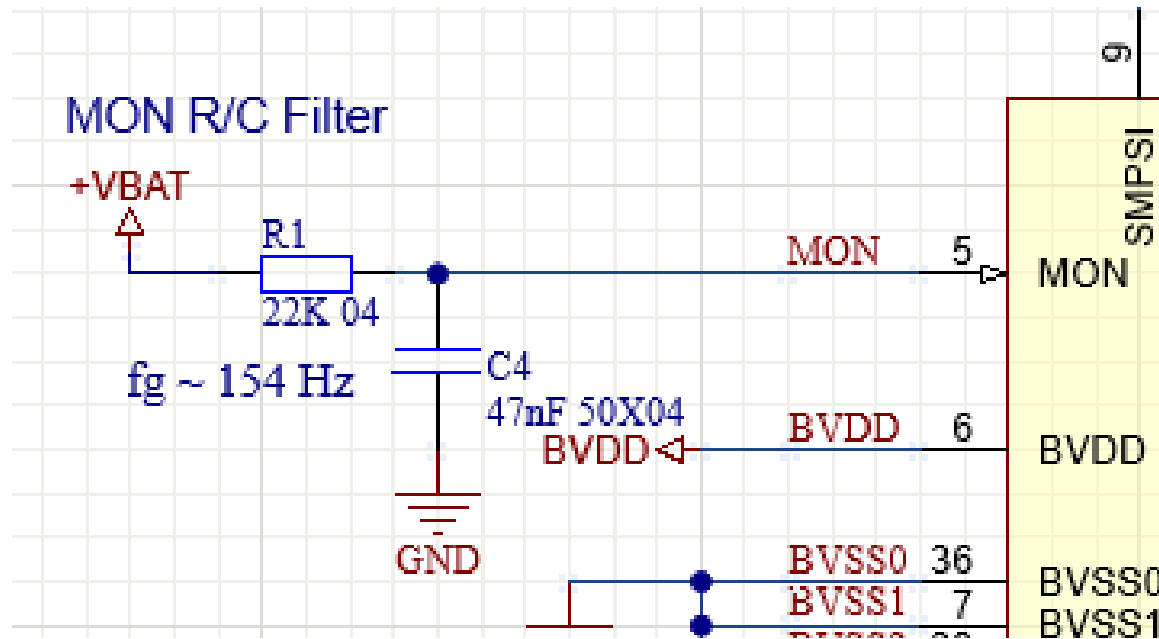
Beispiel:

$$f_{\text{abast}} \approx 2,2 \cdot f_{\text{max}}$$

Auf einer **CD** wird ein Signal gespeichert, das durch die Digitalisierung eines analogen Audiosignals mit Frequenzen bis 20 kHz erzeugt wird. Die Frequenz, mit der das analoge Audiosignal abgetastet wird, beträgt 44,1 kHz.

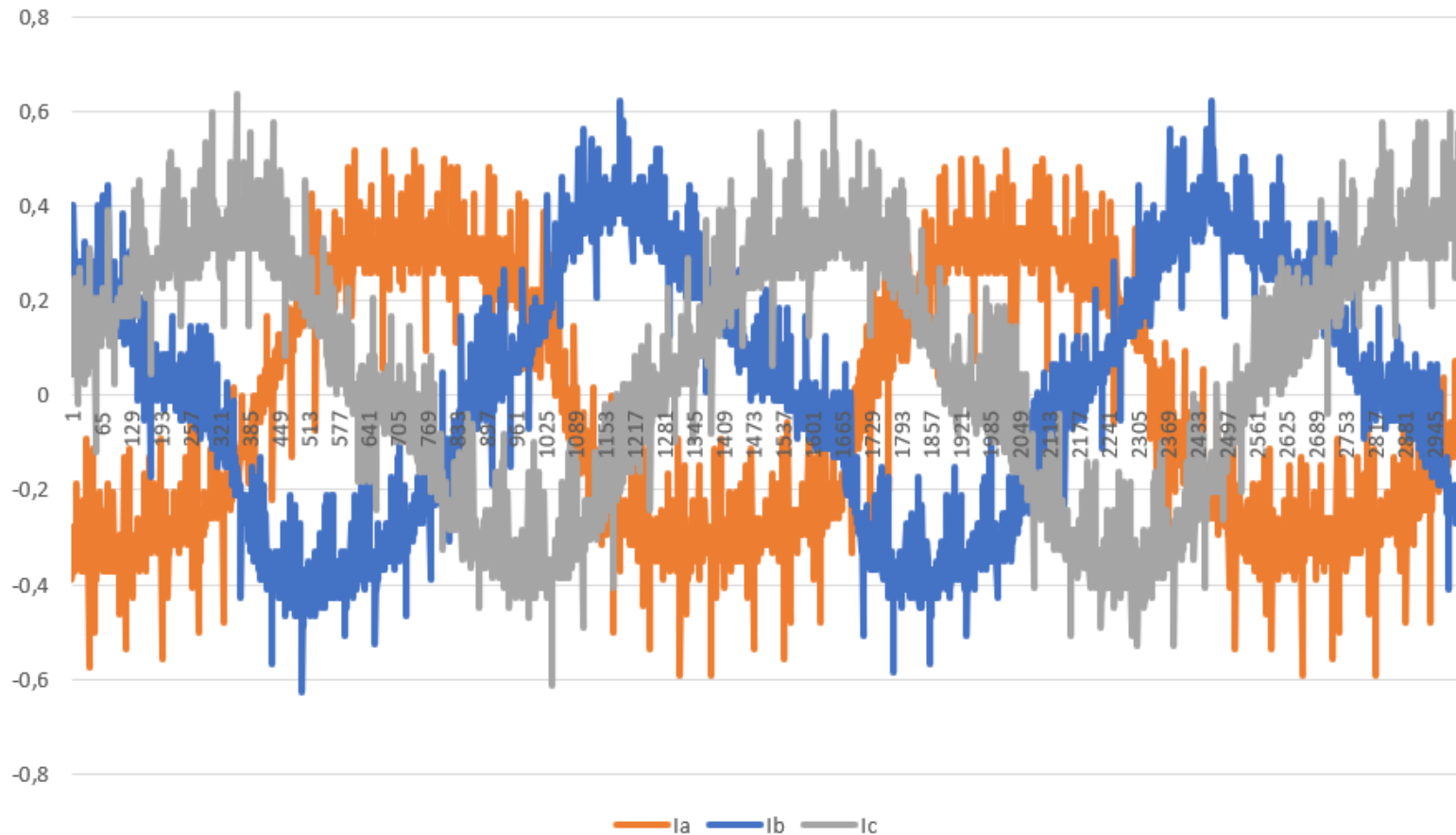
Der verwendete Faktor ist abhängig vom verwendeten Tiefpassfilter und von der benötigten Dämpfung der Alias-Signale. Andere gebräuchliche Faktoren sind 2,4 (**DAT**, **DVD**) und 2,56 (**FFT-Analysatoren**).

## Anti-Aliasing Filter



## Messung von Phasenströmen eines BLDC Motors

Raw Phase Currents 13V/1.3Nm@23°C

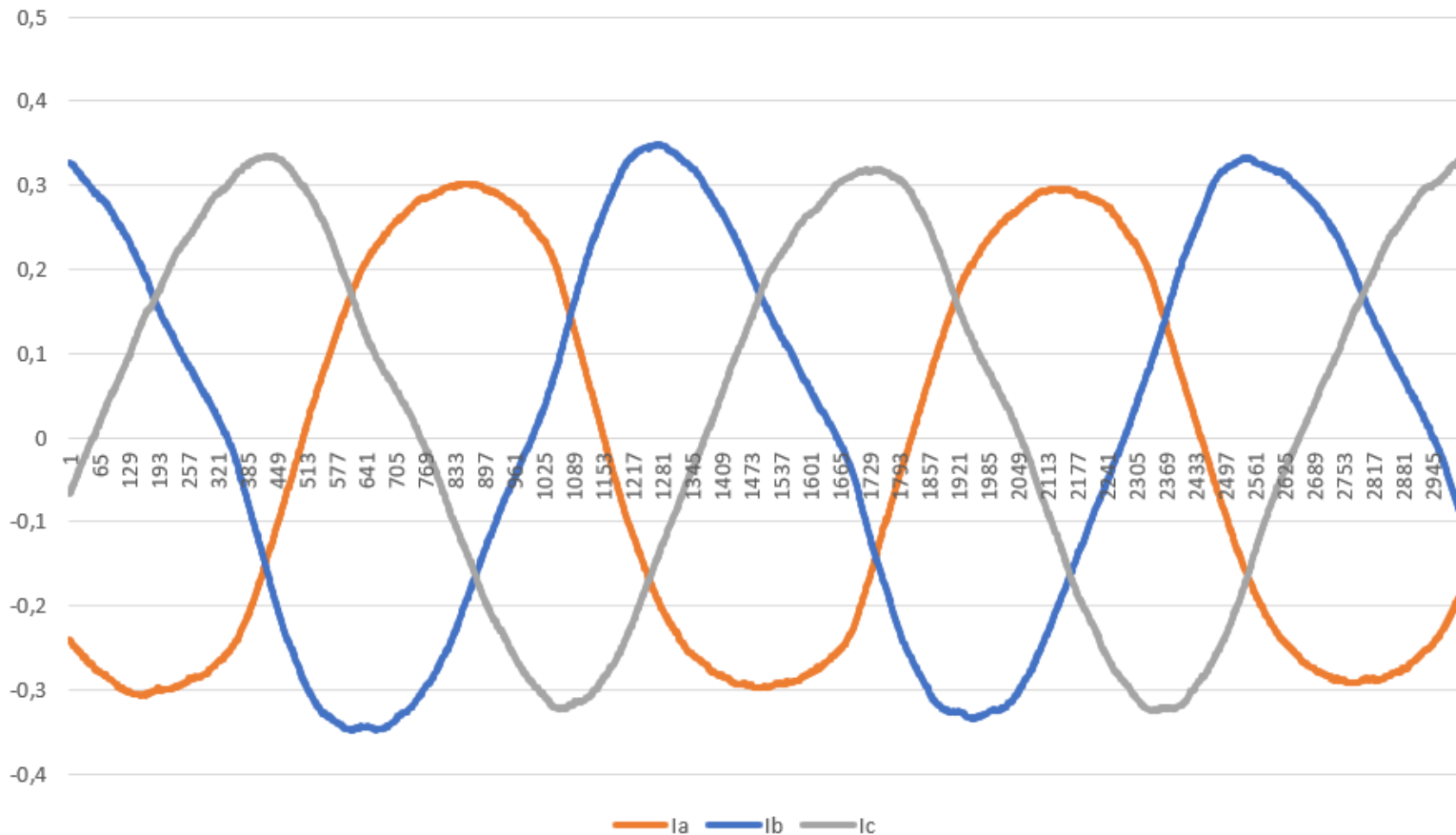


Quelle: intern



## Messung von Phasenströmen eines BLDC Motors

Filtered Phase Currents 13V/1.3Nm@23°C

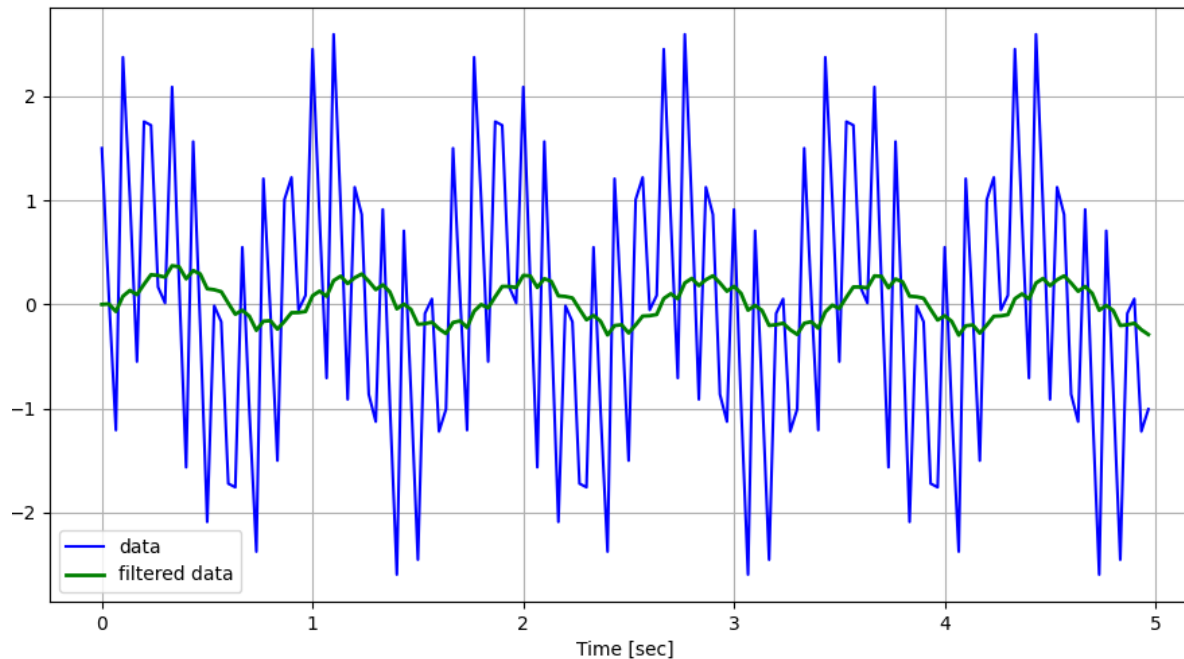


Quelle: intern

## SMA und EMA Filter

<https://blog.mbedded.ninja/programming/signal-processing/digital-filters/windowed-moving-average-filters/>

<https://blog.mbedded.ninja/programming/signal-processing/digital-filters/exponential-moving-average-ema-filter/>



## SMA Filter

A left-handed simple moving average filter can be represented by:

$$y[i] = \frac{1}{N} \sum_{j=0}^{N-1} x[i - j] \quad (2)$$

where:

$x$  = the input signal

$y$  = the output signal

$N$  = the number of points in the average (the width of the window)

## EMA Filter

The *difference equation* for an exponential moving average filter is:

$$y[i] = \alpha \cdot x[i] + (1 - \alpha) \cdot y[i - 1] \quad (1)$$

where:

$y$  is the output ( $[i]$  denotes the sample number)

$x$  is the input

$\alpha$  is a constant which sets the cutoff frequency (a value between 0 and 1)

## Floating Point vs. Fixed Point

Bei Verwendung einer Hardware **ohne FPU**, stellt die Nutzung von Floating Point Daten häufig eine Herausforderung dar. Eine Berechnung mit der sog. SoftFPU ist in der Regel sehr langsam und gerade bei der Signalverarbeitung (z.B. Filter, Regelschleifen etc.) häufig nicht hinnehmbar.

Abhilfe schafft hier die Nutzung von sog. Fixed Point Arithmetik. Hierbei werden rationale Zahlen in Form von Ganzzahlen dargestellt.

### Skalierung mit 10er Basis:

Werte und Berechnungen werden mit einem bestimmten Skalierungsfaktor (zur Basis 10) multipliziert/dividiert

### Skalierung mit 2er Basis: (Q-Format)

Zahlen werden mit einem festen Anteil an Bits für den ganzzahligen Teil sowie für den Nachkommateil interpretiert.

[https://en.wikipedia.org/wiki/Q\\_\(number\\_format\)](https://en.wikipedia.org/wiki/Q_(number_format))

<https://www.allaboutcircuits.com/technical-articles/fixed-point-representation-the-q-format-and-addition-examples/>

Quelle: <https://blog.mbedded.ninja/programming/signal-processing/digital-filters/exponential-moving-average-ema-filter/>

## Entprellen von Digital-Eingänge

