Resolver signal processing

A PMSM motor with only resolver signals are given.
This document meant to examine the ways those signal can be turned into an absolute angle degree, while trying to maintain an accaptable amount of error in the calculation process.

Introduction

A resolver is a rotating transformer with the primary winding in rotor and two secondary windings in stator. The secondary windings are displaced from each other by 90 degrees. Hence, the primary flux linking the secondary windings will be proportional to the sine and the cosine of rotor position. If the primary winding is excited with a voltage $e_p = E_m \sin(\omega_c t)$, then the voltages induced in the secondary windings (sine and cosine) are given by:

$$e_{sine} = k.\ E_m \sin(\omega_c t).\sin(\theta)$$
$$e_{cosine} = k.\ E_m \sin(\omega_c t).\cos(\theta)$$

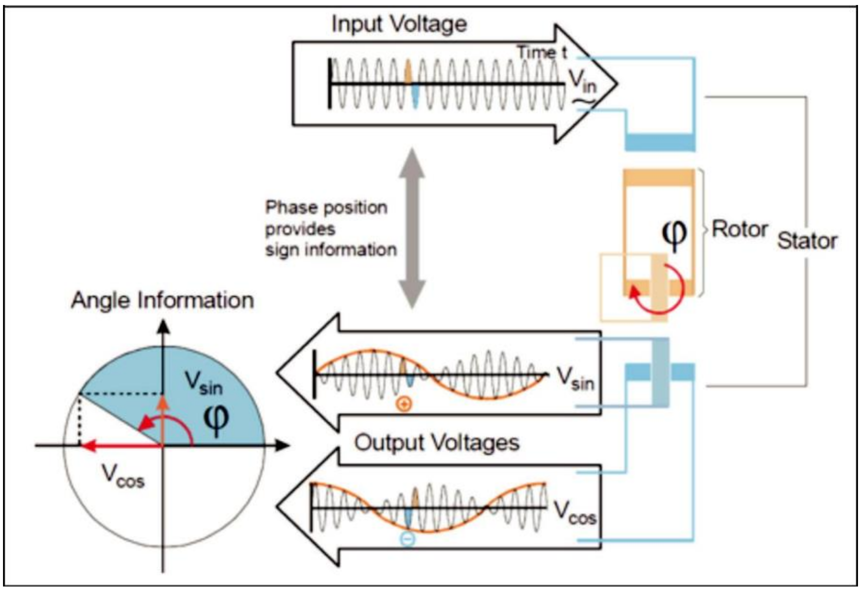
where,

ω_c is the excitation frequency

θ is the rotor position

k is the transformation ratio between primary to secondary

Powering up the primary winding in the rotor is another task. Slip rings, though capable, are not robust. That is why an auxiliary transformer is constructed at one end of the same stator and rotor concentrically, stator holding the primary winding and taking in excitation from external sources and the rotor holding the secondary winding to feed the primary of the resolver transformer as shown in Figure 1.



Since the resolver output signals are trigonometrically related to each other, it is possible to extract rotor position information by a simple math as given by:

$\theta = \tan^{-1}(e_{sine} / e_{cosine}) \rightarrow 1$

The fidelity of such a math is influenced by the quality of parameters in the right-hand side of this equation, which are the secondary voltages. They carry error footprints due to electrical phase shift, transformation ratio mismatch between secondaries and mechanical deviation from quadrature between secondaries. In addition to these native issues, external factors such as ground noise, gain mismatch and nonlinearity in the control circuit and carrier quality issues inject another degree of error. Hence care should be taken in the design of control hardware and software to reject as much of injected error as possible.

Simple method

The most simple and straight forward method is to implement equation 1. It makes sense to sample the values of the secondary signals at their peaks so as to ensure higher signal to noise ratio and use it in equation 1 as shown in Figure 2.

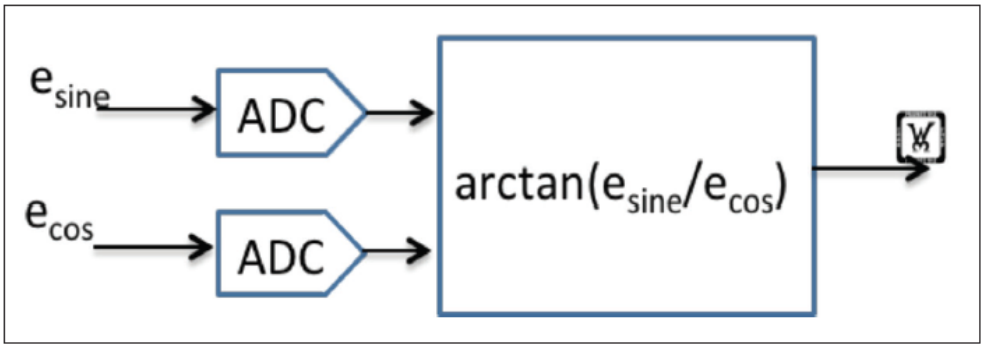


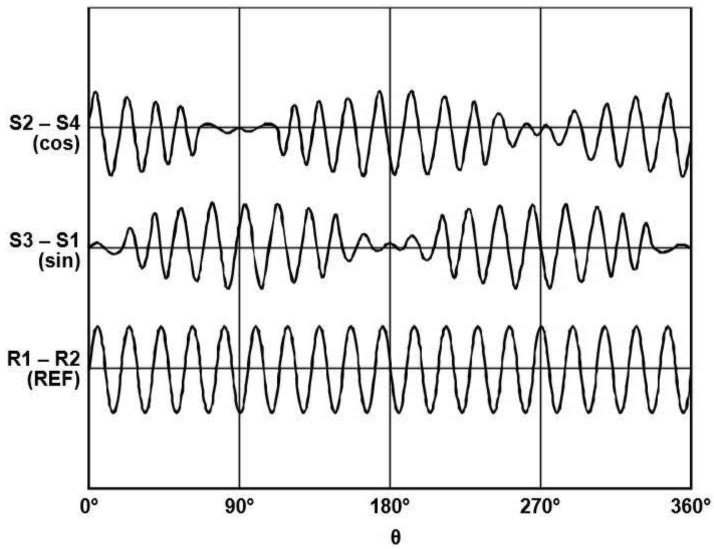
Figure 2: Simple implementation scheme

This simple method lacks in quality, though. The signals might not have ideal scaling and offset (etc.) and due to system noise, the estimated angle will have more jitters, which is why a filter is needed. Unfortunately, a filter will insert phase lag that will linearly vary with speed render-ing itself unsuitable.

So, another option is to correct some errors in the signals before processing, and those are:

- DC offset on the single ended signals (these may not cancel each other completely during the subtraction step),
- amplitude mismatch between the channels and also to the expected value (the following processing stages may expect properly scaled signals),
- offset error on the modulating signals (meaning that the average of the sin(θ) or cos(θ) modulating signal is not zero over a complete rotation, this is a resolver error),
- angular mismatch between the two demodulated sense signals (sine and cosine are not spaced 90° apart, this will not be discussed here),
- harmonics in the two demodulated sense signals (this will not be discussed here).

The ideal behavior of each signal are shown below:



It can be seen, that the excitation signal (REF) is a continuous sine wave with fixed amplitude. The two output signals (sin and cos) are both amplitude modulated versions of the excitation signal (with some additional phase shift).

The equations above show the **ideal** behavior of each signal. The excitation signal is just a free running sine signal with a given amplitude (A) and frequency (ω). The sense signals have a static coupling factor (K) from the excitation signal and a dependency on the rotor angle θ. This corresponds to double-sideband suppressed-carrier modulation, which cannot be demodulated just by envelope detection, the phase difference between the excitation and the sense signals has to be taken into account.

Auxilary signal processing

DC offset correction/tracking

There are multiple options to deal with DC offset (coming from the ADC and sense circuit inaccuracies):

- no DC offset correction, causes errors during demodulation, but as long as the errors are

$$S2 - S4 = K * REF * \cos(\theta)$$
$$S3 - S1 = K * REF * \sin(\theta)$$
$$REF = A * \sin(\omega t)$$

  small this is fine (effect will be shown later)
- end-of-line calibration (may be used to remove static errors, but temperature and lifetime effects will be not removed),
- digital highpass filtering the signals, which would add additional delay to the processing loop, increasing the angle error at high speeds,
- digital lowpass filtering the signals (over significant amount of excitation periods) and then subtracting the lowpass filtered signal from the input signal (this way there is no additional delay in the processing loop),
- filtering the DC offset in the demodulation filter (which would mean stricter requirements for that filter, and probably cause additional delay as well.

- **Modulating signal amplitude and offset correction/tracking**

The modulating signals may have offsets and amplitude errors, which should be corrected before processing the signals any further. There are multiple options to deal with these errors as well:

- no amplitude and offset correction of the modulating signals (as long as the errors are small the angular error will be also small, the effects of these errors will be shown later),
- end-of-line calibration (calibrate end-of-line and then use those coefficients throughout the lifetime of the product),
- peak search on the raw signals to determine the positive and negative amplitude and from those the necessary correction factors (if the signals have DC offset, then that will influence the results and with the peak search the true amplitude may not be found correctly for a motor rotating at high speeds),
- peak search on lowpass filtered demodulated signal outside of the processing loop (no additional delays due to filtering inside processing loop)
- peak search on lowpass filtered demodulated signal inside the processing loop (same as above, but the next stages also using the lowpass filtered signal, this has drawbacks due to the filtering inside the processing loop, but not for the amplitude and offset tracking)

Signal processing options

The majority of applications using resolvers is using one of these techniques to extract the demodulated sense signals of the resolver:

- timed (or "critical") sampling of all signals at the peaks of the excitation signal (which means basically two samples per excitation period, which is demodulation by sampling), it is also called undersampling,
- oversampling of the resolver signals and using these samples to create the demodulated sense signal, these would be some options on how to deal with the sampled signals:
    - peak search on the oversampled excitation signal to determine which sense signal samples to use (which is basically equivalent to the critical sampling and therefore probably wasteful because it throws away samples and it only provides two values each excitation period),
    - peak search and then function fitting to determine the true peak location (which may be numerically intensive and it still provides only two samples per excitation period),
    - demodulating the sense signals by multiplying them with the excitation signal and using all values in the later stages (this could use much of the available samples, except of course where the signals have a zero crossing and the demodulated signal has a significant amount of harmonics which the later stages have to deal with)
    - demodulation by multiplication followed by lowpass filtering (eliminates the harmonics in the demodulated signal at the expense of introducing a time delay and therefore angle error at high speeds)

Only four methods are investigated in this document:

- undersampling
- peak search oversampling (without function fitting) which, with properly selected parameters is equivalent to critical sampling
- demodulation without filtering
- demodulation with filtering

Resolver-to-Digital (R/D) Conversion – using Undersampling

The basic method is depicted in Figure 3. The sine and cosine modulated output signals **u_1 and u_2 must be sampled at the same frequency as the reference frequency.** This, so called **undersampling**, demodulates both analog signals, so that the digitized samples $u_1(n)$ and $u_2(n)$ are $\sin\varepsilon$ and $\cos\varepsilon$ of the angle ε, respectively. This method can be ideally implemented on the TMS320F240. It incorporates dual ADCs, which can be synchronized to the reference frequency, generated by the on-chip PWM-unit.

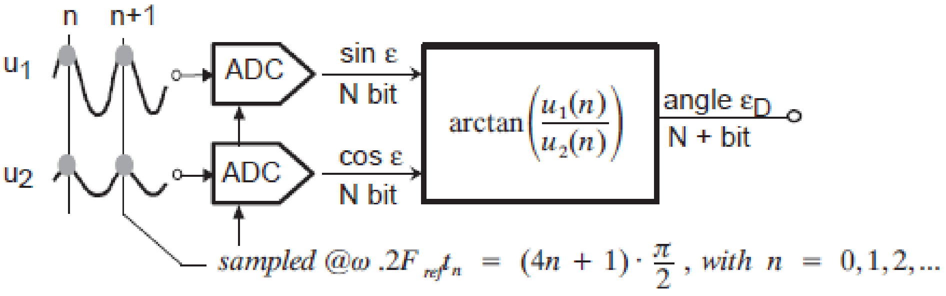


Figure 3. Resolver-to-Digital Conversion Utilizing Undersampling and Inverse Tangent

The angular position can now be determined by a four quadrant inverse tangent function of the quotient of the demodulated sine and cosine samples. The inverse tangent function is ambiguous. Thus, the sign of the sampled signals has to be taken into account, in order to determine the absolute angular position according to:

$$\varepsilon(n) = \begin{cases} \arctan\left(\dfrac{u_1(n)}{u_2(n)}\right), & \text{if } u_2(n) \geq 0 \\ \pi + \arctan\left(\dfrac{u_2(n)}{u_1(n)}\right), & \text{if } u_2(n) < 0 \end{cases}$$

To be accurate, both signals, u_1 and u_2, have to be sampled simultaneously, at, or close to their maximum positive value, synchronized to the reference frequency according to:

$$\omega_{ref}t_n = (4n+1)\frac{\pi}{2},\ n = 0, 1, 2, \ldots,$$

To avoid aliasing, the Nyquist criteria must be met. It requires the sample rate f_S to be at least twice the *bandwidth* f_B of the interesting signal. To meet that, an analog anti-alias filter has to remove any frequency components outside the band-of-interest $f_{ref} \pm f_B$. Referring to Figure 2, it is obvious that any DC offset has to be removed prior to sampling. Otherwise, it would be added to the demodulated sine and cosine signals and decrease accuracy.

For an N-bit ADC, the angular accuracy achievable is N+1 bit. A higher resolution and better noise rejection are achievable by oversampling and averaging techniques, which are discussed in the following section.

Peak search with oversampling

This method determines the maximum and minimum locations on the excitations signal and then extracts the sine and cosine signal values from the sense signals at the same time instants. (At the minimum location the sine and cosine signals are inverted.)

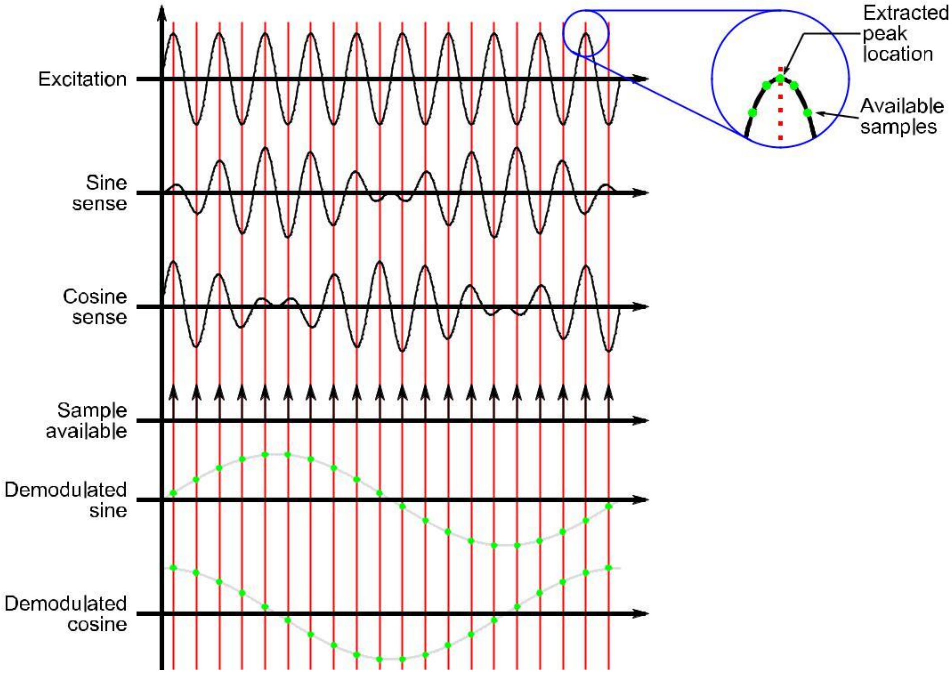


Figure 3 Peak search demodulation

The figure above shows the input and output signals of the peak search algorithm. As it can be seen demodulated **samples are available twice in each period** and the exact peak location is determined based on the excitation signal. The sense signal values at the same moment are then forwarded to further processing, all the other samples are thrown away.

Demodulation without filtering

This method basically multiplies the excitation and the sense signals together to get a demodulated signal (with significant harmonic content) to be fed into the tracking loop.

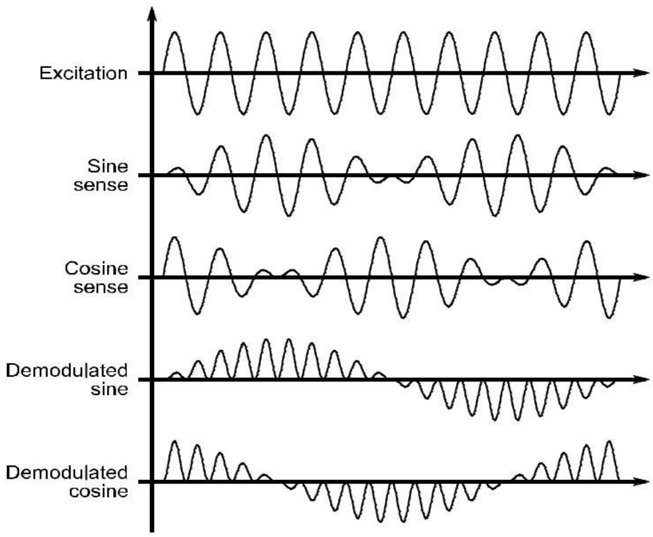


Figure 4 Demodulation without filtering

The figure above shows the inputs and outputs of the demodulation algorithm. (As already mentioned the output signals have significant harmonic content). The advantage of this demodulation is, that all input samples are processed and can be fed into the tracking loop. (The tracking loop has then to deal with the harmonics.)

Demodulation with filtering

In this method the excitation and sense signals are multiplied together and the lowpass (or a bandpass) filtered to get a demodulated signal with some phase shift.

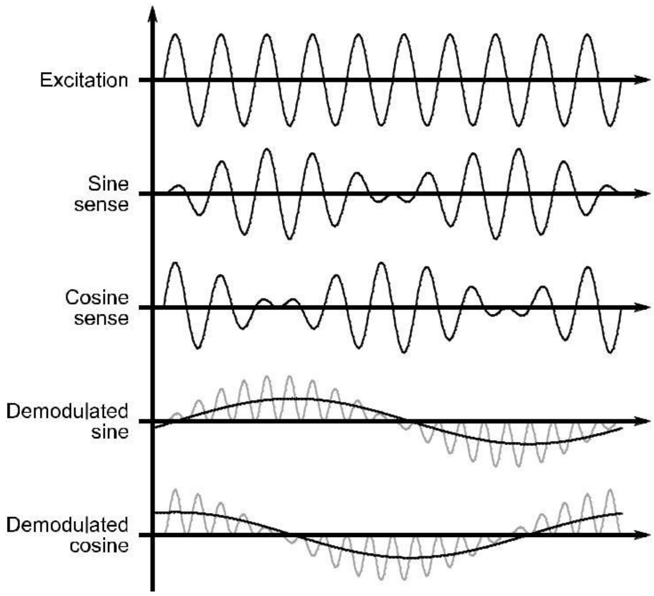


Figure 5 Demodulation with filtering (lowpass)

The figure above shows the inputs and outputs of this demodulation algorithm. In light grey the input signals to the filters are shown and the filter output is shown in black. Due to the filtering there is some phase shift between the true sine and cosine signals and the filtered sine and cosine signals. Since the harmonics are already filtered out the tracking loop gets a more stable input signal, but with additional phase shift, which will then show up on its output as angle error.
So a solution is to handle that angle error in the control loop itself (below with oversampling):

R/D Conversion – Improved Method Using Oversampling

Overview

The total algorithm, utilizing oversampling, is depicted in Figure 4. The high-resolution digital angle ε_m is the output of the position-tracking closed-loop.

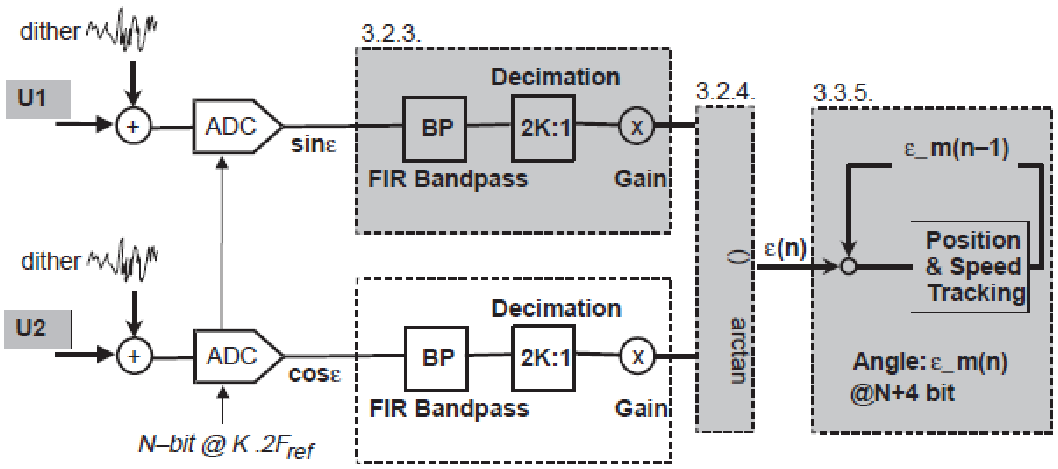


Figure 4. Block Diagram of the Improved Resolver-to-Digital Converter

The sine and cosine modulated resolver output signals u_1, u_2 are sampled at 2K-times the reference frequency. This is equivalent to K-times oversampling. **Dither** is added prior to sampling, to ensure that the quantization noise is not correlated to the resolver output signals.

The **FIR bandpass filter** acts as a digital anti-alias filter and improves the resolution of the samples by reducing the bandwidth to $f_{ref} \pm (f_{ref}/2)$. The decimation is done taking only every $2K^{th}$ sample. This demodulates the bandpass-filtered samples and reduces the sample rate to f_{ref}. At that point, the (filtered) sine and cosine samples of the resolver angular position are available.

The angular position is now derived by the **arc tangent** of the quotient of the demodulated sine and cosine samples. Due to the averaging FIR filter, the resolution of the angular position has been improved. On the other hand, it shows a velocity lag.

The **position and speed interpolation closed loop** is added to further improve the resolution of the angular position. The achievable improvement depends on the bandwidth of the closed loop. Another important task of the closed loop is that it exactly compensates the group delay of the FIR bandpass filter. Thus, the estimated angle ε_m shows a higher resolution, but does not suffer any velocity lag.

Dither – Adding Random White Noise

To apply oversampling or averaging techniques, the **quantization error e(n) must not be correlated with the input signal**. This is obviously not true for example, for DC signals, where the quantization error e(n) remains constant. In that case the quantization error is correlated and averaging can not improve the resolution.

For example, a solution to that problem is to add random white noise to the analog signal, if noise is not already present. The additive noise must have a root-mean-square value of

$$rms(noise) = \sqrt{q/12}, \quad q = 1LSB$$

This ensures that the quantization error is not correlated with the input signal. Note that this does not increase the total quantization error.

FIR Decimation Bandpass-Filter

The decimation bandpass filter is a symmetrical FIR filter, which has got a linear phase. The number of taps depends mainly on the oversampling ratio. The center frequency of the FIR filter is set equal to the resolver reference frequency, as shown in Figure 5.

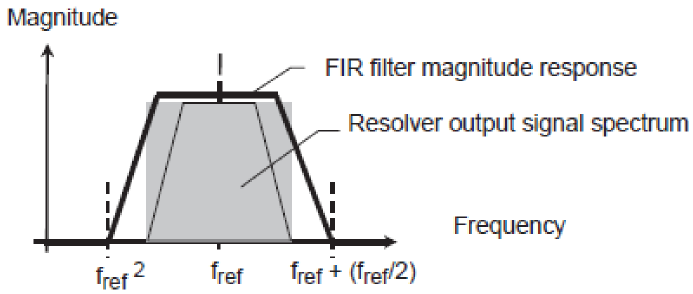


Figure 5. FIR Decimation Bandpass Filter Magnitude Response

The task of the FIR filter is to reduce the bandwidth to [$f_{ref} \pm (f_{ref}/2)$]. This is required prior to decimation to fulfill the Nyquist criteria. The FIR filter is, in fact, a digital anti-alias filter, which additionally improves the resolution within the band-of-interest. For K-times oversampling, decimation in time is achieved by taking only every 2Kth output of the FIR filter. That is equivalent to calculating the filter for only every 2Kth input sample. Decimation in time will demodulate the signals down to the base-band [0-f_{ref}/2].

Then the sine and cosine of the resolver angular position are at a higher resolution. As a rule of thumb the resolution increases by 3dB (equivalent to 0.5bit), each time the bandwidth is halved.

On the other hand the signals are delayed. This is due to the constant group delay of the symmetrical FIR filter. For N taps, the group delay is exactly (N-1)/2 samples. When the FIR filter has got (4K+1) taps, the group delay is equivalent to one period of the reference frequency.

Arc Tangent Function

The angle is derived by taking the arc tangent of the quotient of the FIR filtered sine and cosine signals, as outlined in section 3.1.

$$\varepsilon_{FIR}(n) = \begin{cases} \arctan\left(\dfrac{u_{1,FIR}(n)}{u_{2,FIR}(n)}\right), & u_{2,FIR}(n) \geq 0 \\[2ex] \pi + \arctan\left(\dfrac{u_{2,FIR}(n)}{u_{1,FIR}(n)}\right), & u_{2,FIR}(n) < 0 \end{cases}$$

Compared to the basic method, shown in section 3.1, the digitized angle now has got a **higher resolution**, due to the averaging FIR filter. However, as depicted in Figure 6, the digitized angle shows a velocity lag, due the group delay of the FIR filter. The error of the digitized angle is proportional to the rotational speed.

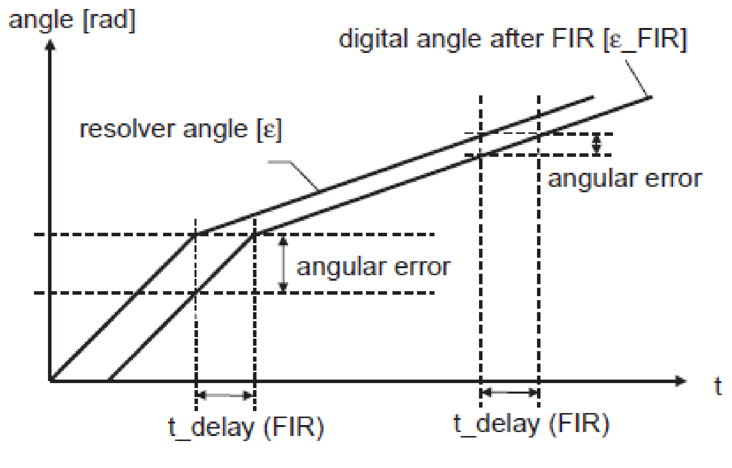


Figure 6. Angular Error as Function of the Rotational Speed After the FIR Filter

A closed-loop position and speed interpolator will be used to compensate for the angular error and to further improve the angular resolution.

Closed-Loop Angular Position and Speed Interpolator

The closed-loop, shown in Figure 7, consists of a PI controller, a 1st order IIR filter and an integrator. It is basically a lowpass filter. It is a type 2 closed-loop (two integrators) and hence does not have an integral error. The parameters of the closed-loop define bandwidth and characteristic and are discussed in the next section.

The task of the closed-loop is to:

- Improve the accuracy of the angular position ε_m. The accuracy depends on bandwidth selected.
- Compensate the delay of the FIR filter, so that ε_m suffers no velocity lag
- Derive the angular speed ω_m

The result will be a **higher accuracy** of the interpolated angular position ε_m. Additionally ε_m **does not suffer a velocity lag.**

This is achieved by the following. The output signal ε_m is delayed with the same time as the FIR filters group delay. Hence the delayed output signal ε_m(n–1) is also compared with the delayed FIR filtered angle ε_FIR. The closed-loop forces both delayed angles to be identical. For constant speeds, the output ε_m of the closed-loop is then identical with the real resolver shaft angle.

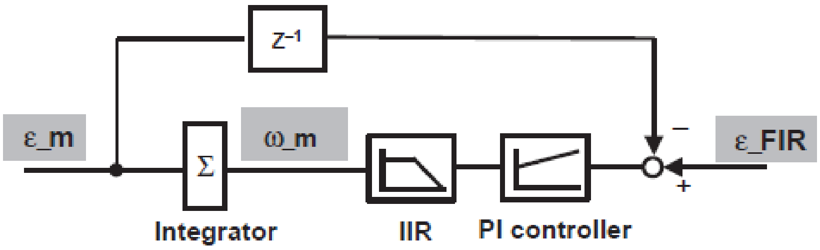


Figure 7. Closed-Loop Position and Speed Interpolator

So in conclusion all that need to be implemented in order for the resolver to work with a microcontroller are:

- DAC config which outputs the exciting sin waveform
- An amplifier which amplifies this signal for the resolver input (stator coal)
- The errors of the resolver's output-signals can be filtered before processing them, also the outputs need to be scaled to the inputs of the microcontroller
- Now the ADCs need to be configured and the signals need to be demodulated
- After that, the easiest option is to just simply calculate the angle with an arctan function
- If that is not enough, a control loop can be implemented to further refine the measurement