Measurement of Resolver Position

using Genesis HighSpeed Data Recorders

Authors

E. Armando, R. Bojoi
Power Electronics Innovation Center
Dipartimento Energia, Politecnico di Torino

HBM Test and Measurement

Measurement of Resolver Position using Genesis HighSpeed Data Recorders

Summary

The resolver is an electrical machine that is intensively used in synchronous motor drives to measure the rotor position that is needed for the motor control algorithm. Due to its robustness, the resolver is the popular choice in harsh environments, where the electrical motor is subjected to mechanical stresses and vibrations, such as automotive and aerospace applications. The measured position is usually obtained through a resolver-to-digital converter that implements in a single chip several blocks, such as a signal demodulator, a voltage-controlled oscillator and a tracking loop. This work presents a method to obtain the position measured by a resolver using the high-speed voltage acquisition channels of HBM Genesis HighSpeed data recorders, with no requirement of additional hardware.

1. Introduction

The resolver is a small electrical machine used as an absolute position sensor. The stator contains two distributed windings whose magnetic axes are orthogonal, while the rotor contains one distributed winding. In most cases, the pole-pairs number of the stator and rotor windings is equal to unity (p=1). Therefore, in this work the mechanical rotor position is considered equal to the electrical rotor position.

The rotor is fed with sinusoidal voltage supply with frequency (called excitation frequency f_{ecc}) in the range of [2, 10] kHz. A brushless resolver uses a rotating transformer to supply the rotor, as shown in Fig.1 and Fig. 2.

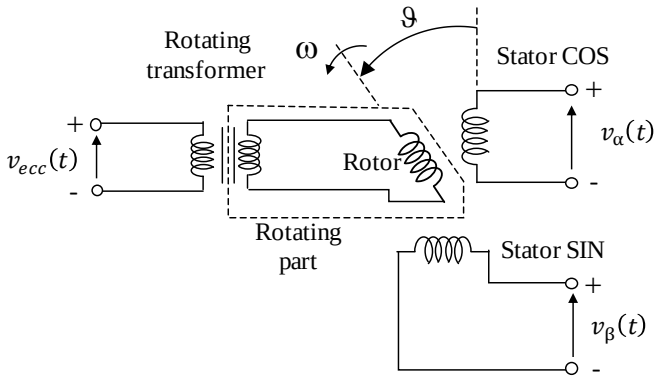

Fig. 1: Electrical schematic of brushless resolver with rotating transformer for rotor supply.

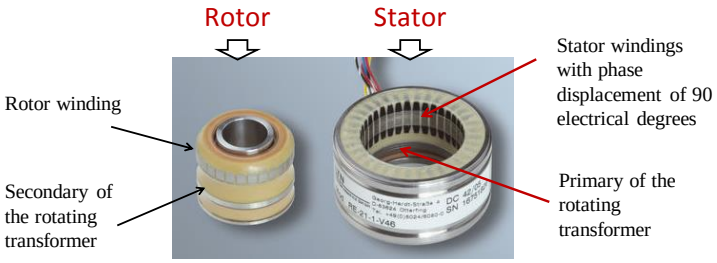

Fig. 2: View of the rotor (left) and the stator (right) of a brushless resolver.

The primary of the rotating transformer is fed by an excitation voltage of given amplitude V_{ecc} and frequency f_{ecc} as

$$v_{ecc}(t) = V_{ecc} \cdot \sin(\omega_{ecc} \cdot t) \tag{1}$$

where $\omega_{ecc} = 2 \cdot \pi \cdot f_{ecc}$ is the pulsation of the excitation voltage. Both V_{ecc} (V) and f_{ecc} (Hz) are defined by the resolver data sheet. To get satisfactory results, the excitation frequency f_{ecc} must be much higher than the mechanical frequency f_m, i.e. $f_{ecc} > f_{mech}$.

The magnetic coupling between the rotor winding and the stator windings depends on the rotor position ϑ. As a result, the voltages that are induced in the stator windings are

$$v_{\alpha}(t) = K \cdot \cos\vartheta \cdot \sin(\omega_{ecc} \cdot t) \tag{2}$$

$$v_{\beta}(t) = K \cdot \sin\vartheta \cdot \sin(\omega_{ecc} \cdot t) \tag{3}$$

where K depends on the coupling factor between stator and rotor and on the turns ratio of the rotating transformer.

From (2) and (3) it results that the voltages at the output of the stator windings contain information on the rotor position. An example of stator output voltages (pu) is shown in Fig. 3 for a constant mechanical speed of $\omega_m = 2 \cdot \pi \cdot 50$ (rad/s) when the resolver is excited at 1 kHz, i.e. $\omega_{ecc} = 2 \cdot \pi \cdot 1000$ (rad/s).

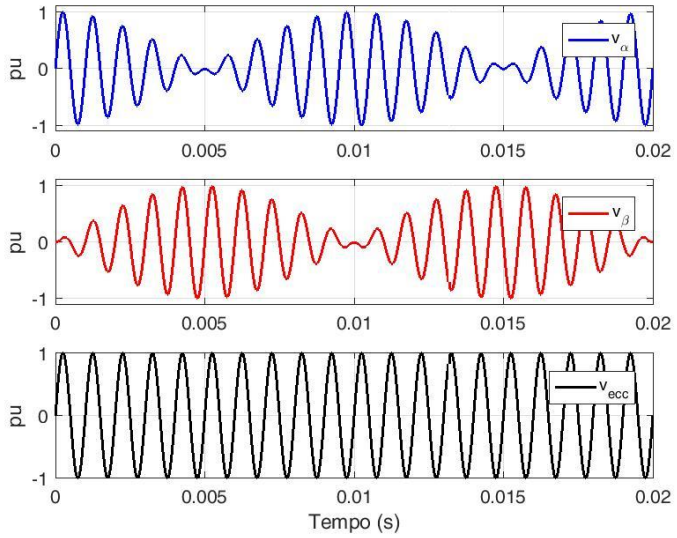

Fig. 3: Steady-state operation at a mechanical speed $\omega_m = 2 \cdot \pi \cdot 50$ (rad/s) with the resolver excited at 1 kHz.

It can be noted that the (α,β) voltages are the cosine and the sine of the rotor position that are modulated with the resolver excitation frequency. Therefore, a demodulation is necessary to remove the excitation frequency, as explained in the next subsection.

2. Resolver Simulation and Signal Elaboration

The resolver has been simulated in Perception using the formulas reported in the Appendix to implement (1)-(3). The resolver excitation frequency has been set at 10 kHz. The resolver excitation voltage v_{ecc} and the resolver output signals $v_{\alpha\beta}$ are shown in Fig. 4 and Fig. 5 for a mechanical speed of 3000 rpm.

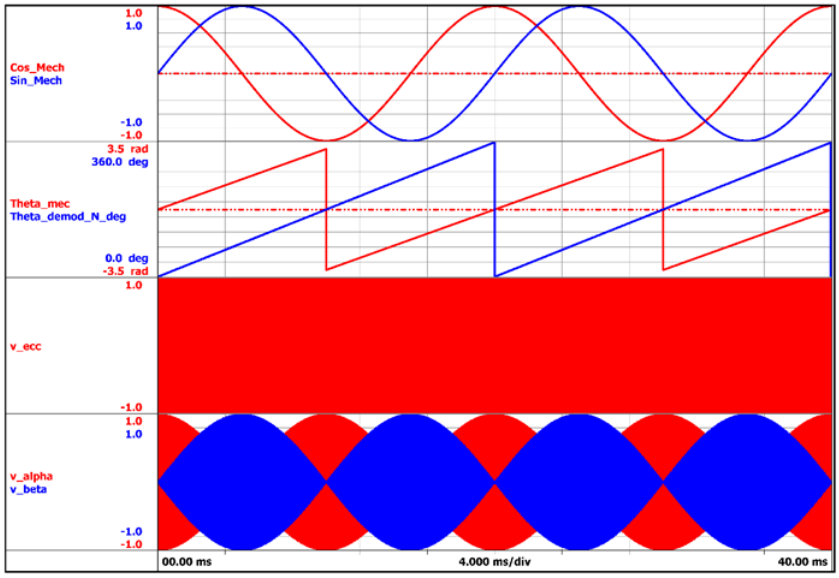

Fig. 4: Simulated steady-state operation at 3000 rpm. From top to bottom: sine and cosine of the rotor position, rotor position, excitation voltage and (α,β) output voltages.

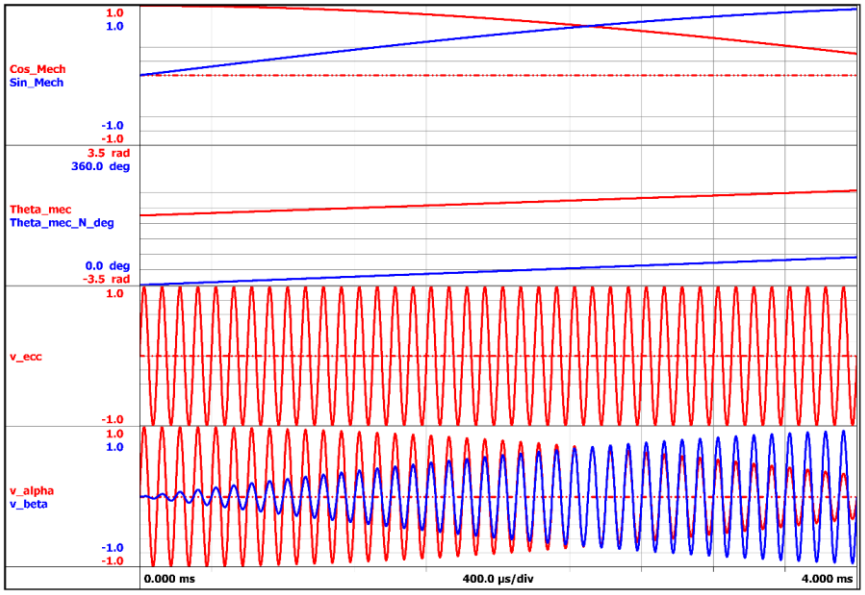

Fig. 5: Simulated steady-state operation at 3000 rpm (zoom). From top to bottom: sine and cosine of the rotor position, rotor position, excitation voltage and (α,β) output voltages.

As mentioned before, the rotor position can be extracted by means of a proper demodulation strategy. For this purpose, the (α,β) output voltages are multiplied with the excitation voltage in (4, 5):

$$v_{\alpha,\mathrm{demod}}(t) = V_{ecc} \cdot K \cdot \cos\vartheta \cdot [\sin(\omega_{ecc} \cdot t)]^2 \quad (4)$$

$$v_{\beta,\mathrm{demod}}(t) = V_{ecc} \cdot K \cdot \sin\vartheta \cdot [\sin(\omega_{ecc} \cdot t)]^2 \quad (5)$$

As can be noted from (4), (5), the signs of $v_{\alpha,\mathrm{demod}}$ and $v_{\beta,\mathrm{demod}}$ are the signs of $\cos\vartheta$ and $\sin\vartheta$, respectively.

The demodulation is performed by calculating the moving average values of the $v_{\alpha,\mathrm{demod}}$ and $v_{\beta,\mathrm{demod}}$ on half of the excitation period $0.5T_{ecc} = 0.5/f_{ecc}$, using the powerful cycle detection function of the Genesis data recorders:

$$cos_{\mathrm{demod}}\,(t) = cycle\ mean\left[v_{\alpha,\mathrm{demod}}\,(t)\right]\Big|_{0.5cycle\ \_ecc} = K' \cdot \cos\vartheta \tag{6}$$

$$sin_{\mathrm{demod}}\,(t) = cycle\ mean\left[v_{\beta,\mathrm{demod}}\,(t)\right]\Big|_{0.5cycle\ \_ecc} = K' \cdot \sin\vartheta \tag{7}$$

It must be noted that the amplitudes of $ccc\mathrm{demod}\,(t)$ and $sss\mathrm{demod}\,(t)$ are not unity, but they should be the same (K'). Therefore, the rotor position can be easily obtained with the Perception function SpaceVectorInverseTransformation that directly provides the result in electrical degrees. The simulated results with Perception in the above described method are shown in Fig. 6-9.

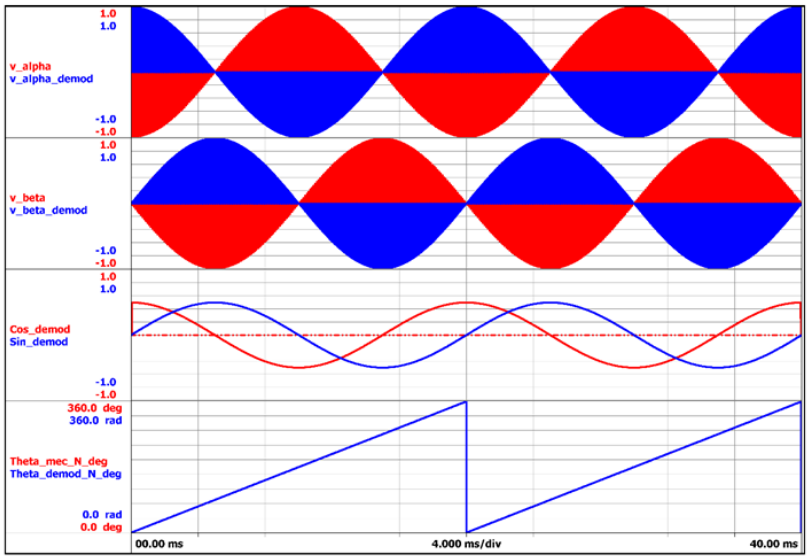

Fig. 6: Simulated steady-state operation at 3000 rpm. From top to bottom: v_{α}, v_{β} output voltages, $v_{\alpha,\,demod}$, $v_{\beta,demod}$, cos_{demod}, sin_{demod}, real position and reconstructed position.

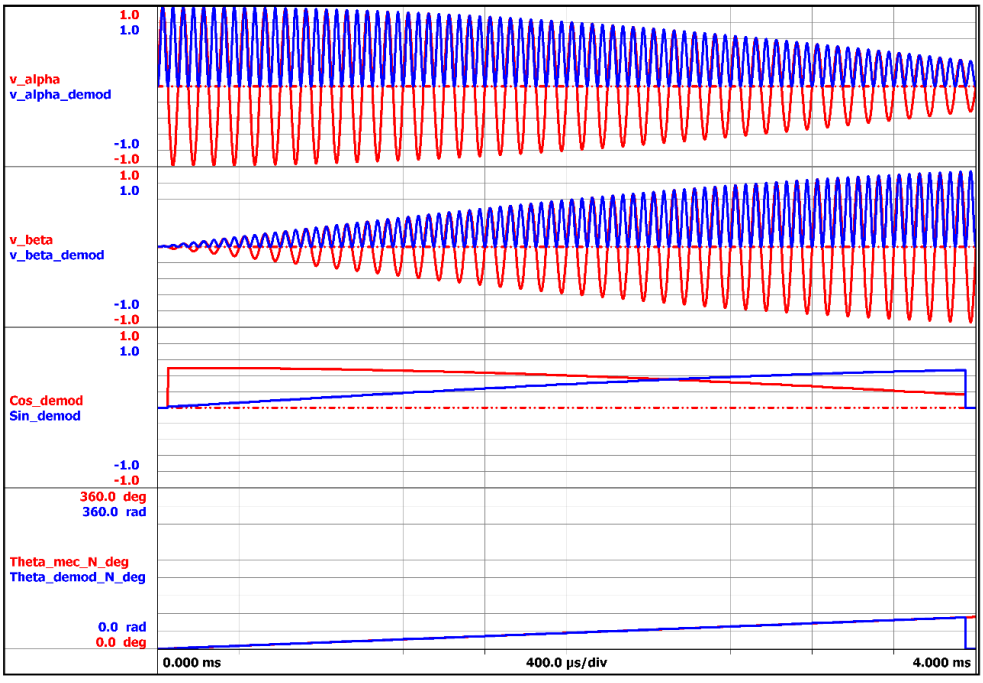

Fig. 7: Simulated steady-state operation at 3000 rpm (zoom). From top to bottom: v_{α}, v_{β} output voltages, $v_{\alpha,demod}$, $v_{\beta,demod}$, $cos_{demod,}$, sin_{demod}, real position and reconstructed position.

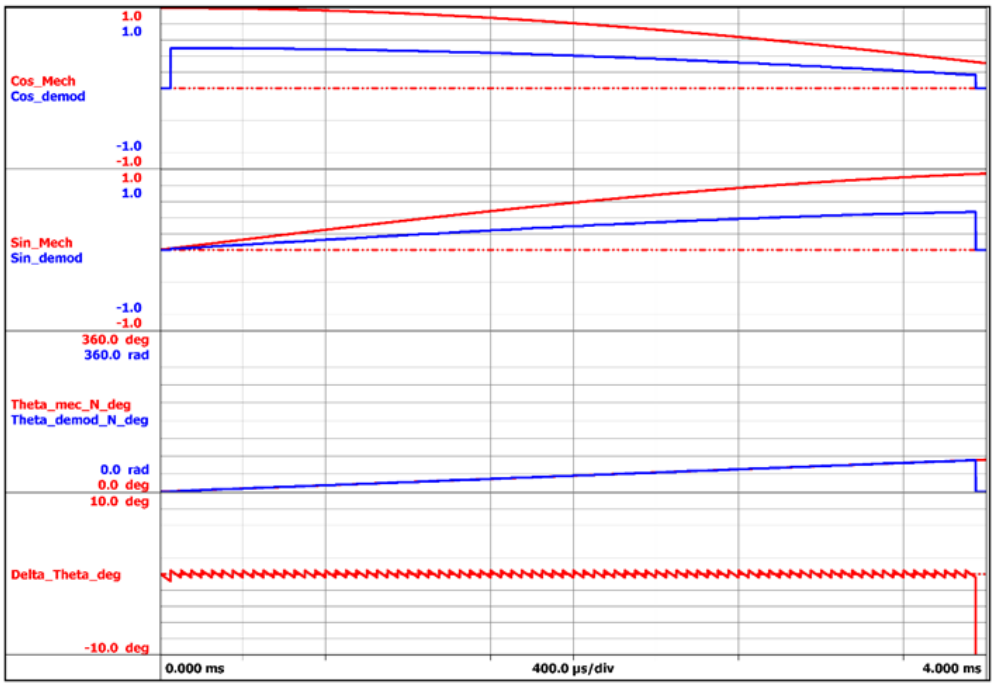

Fig. 8: Simulated steady-state operation at 3000 rpm (zoom). From top to bottom: $cos_{demod,}$, sin_{demod}, real position and reconstructed position, error between real position and reconstructed position.

The error between the real position and reconstructed position is shown in Fig.8. This error exhibits a sawtooth waveform with very low amplitude value (below one electrical degree).

The simulation results for a speed of 18000 rpm are shown in Figs. 9-11. As can be noted in Fig. 11, the error of the position reconstruction increases at high speed, since the position is reconstructed with a fixed sampling rate that depends on the excitation frequency (time-based sampling).

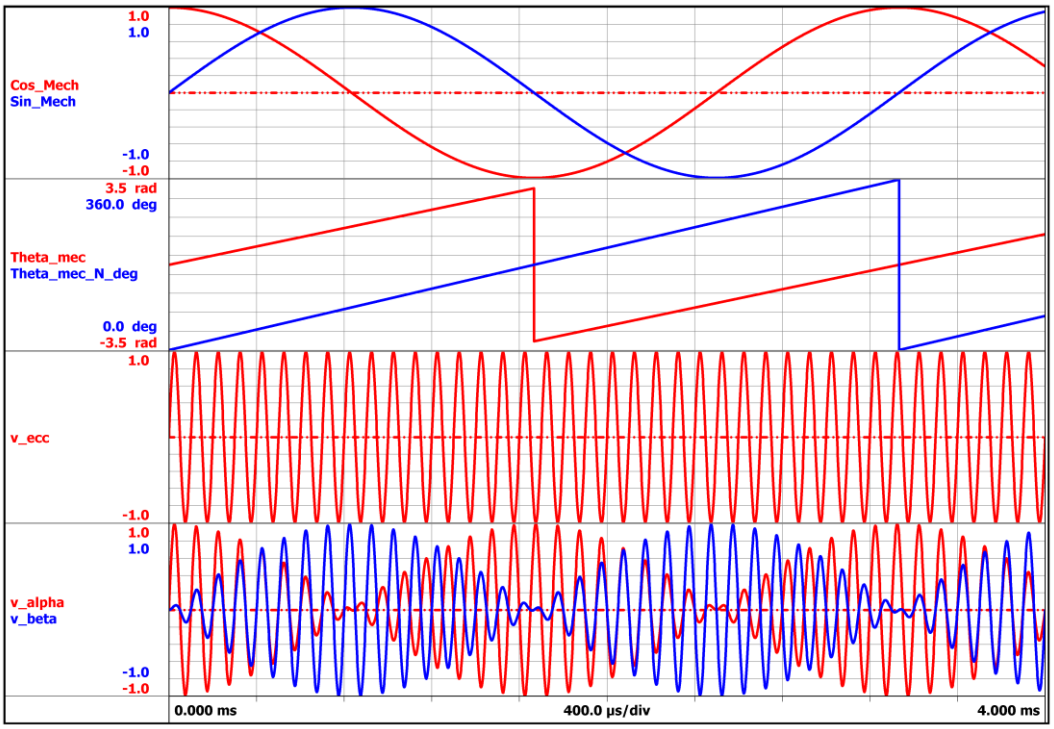

Fig. 9: Simulated steady-state operation at 18000 rpm. From top to bottom: sine and cosine of the rotor position, rotor position, excitation voltage and (α,β) output voltages.

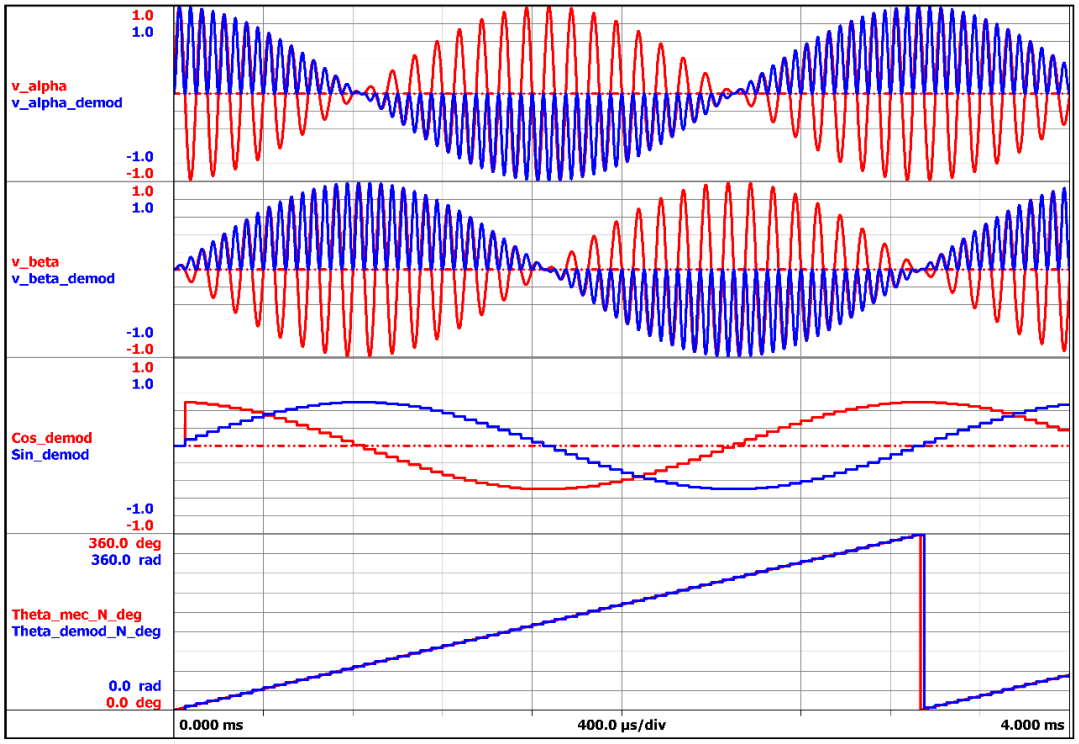

Fig. 10: Simulated steady-state operation at 18000 rpm. From top to bottom: vα, vβ output voltages, $v_{\alpha,demod}$, $v_{\beta,demod}$, cos_{demod}, sin_{demod}, real position and reconstructed position.

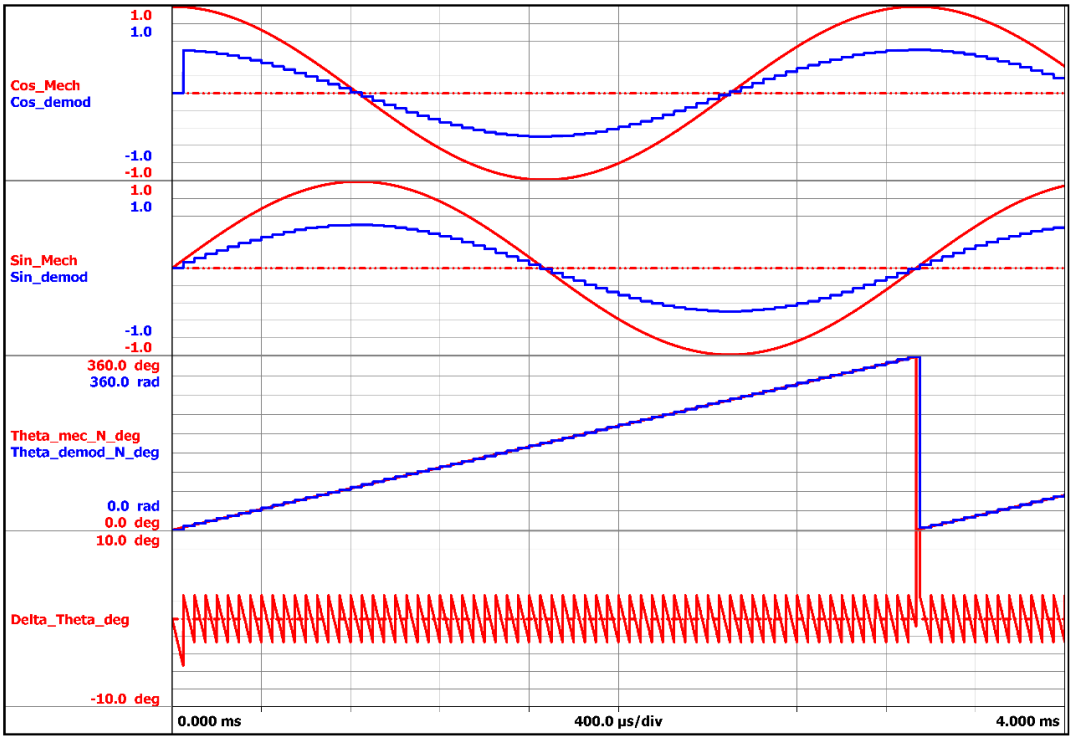

Fig. 11: Simulated steady-state operation at 18000 rpm. From top to bottom: $cos_{demod,}$ sin_{demod}, real position and reconstructed position, error between real position and reconstructed position.

The effects of the acquisition offsets on the (α,β) resolver output voltages have been simulated for two different scenarios:

(1) +2% offset on the α-channel and -2% offset on the β-channel; the results are shown in Fig. 12.

(2) +7% offset on the α-channel and +7% offset on the β-channel; the results are shown in Fig. 13.

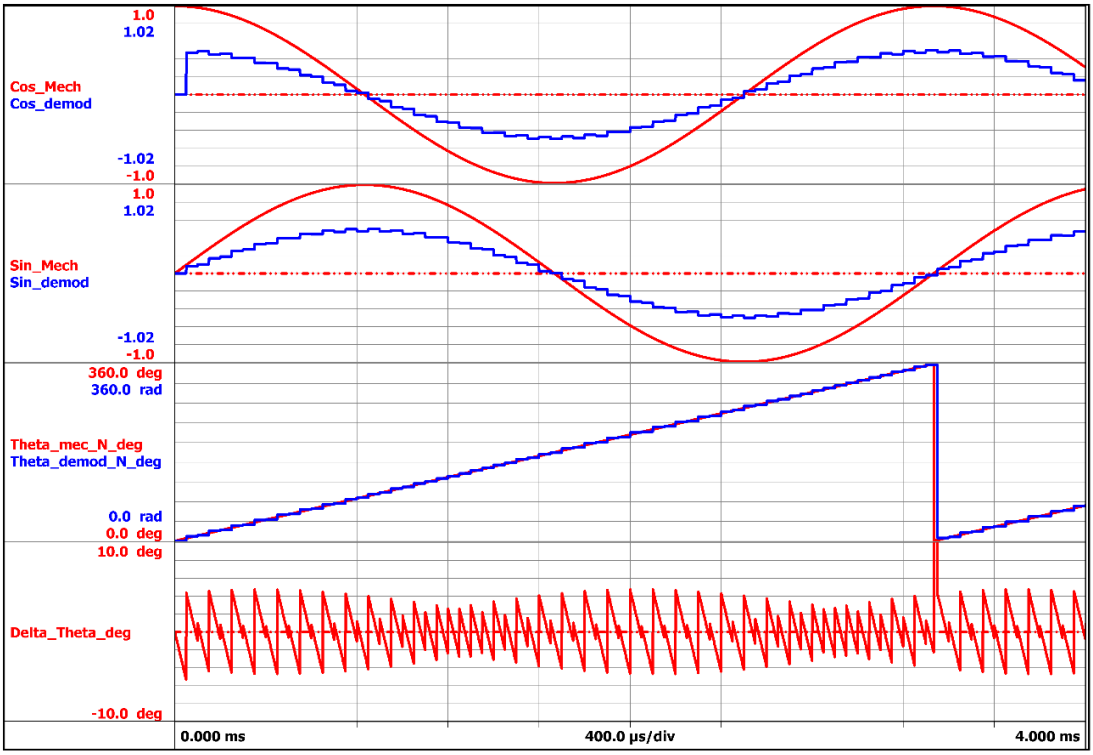

Fig. 12: Simulated steady-state operation at 18000 rpm with offset scenario 1. From top to bottom: $cos_{demod,}$ sin_{demod}, real position and reconstructed position, error between real position and reconstructed position.

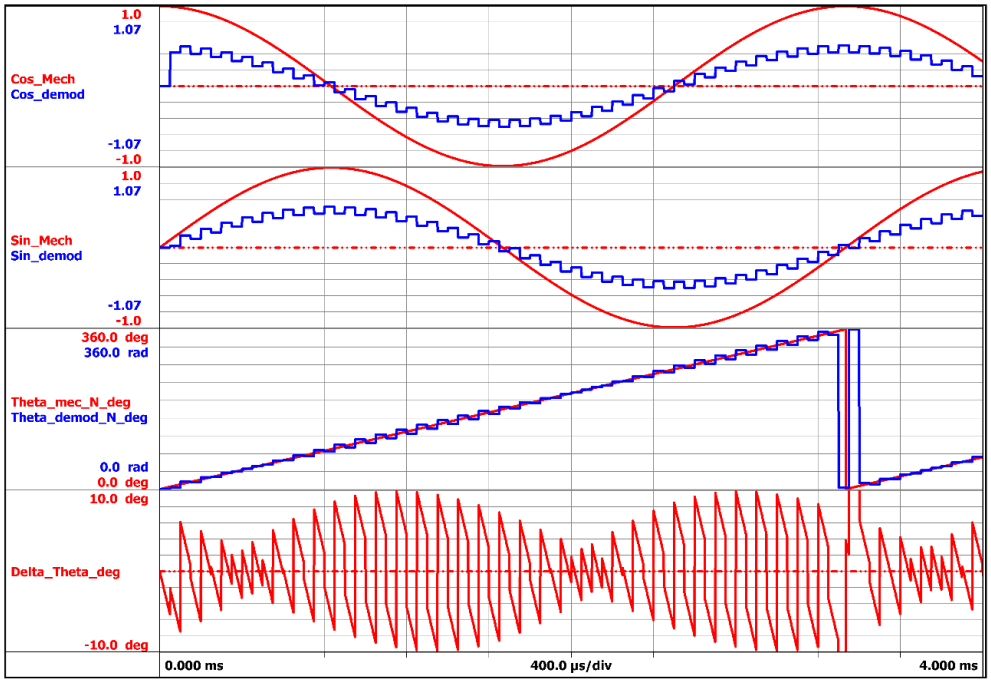

Fig. 13: Simulated steady-state operation at 18000 rpm with offset scenario 2. From top to bottom: $cos_{demod,}$ sin_{demod}, real position and reconstructed position, error between real position and reconstructed position.

The effects of the offsets are quite important, because they may negatively influence the proper reconstruction of the (*d,q*) quantities that are computed by the data recorder. A possible solution is to apply a Low Pass Filter (LPF) on the demodulated sine and cosine function (6) and (7). In this work, a second order Bessel LPF with a cut-off frequency of 1 kHz has been employed:

$$cos_{\text{demod2}}\ (t) = LPF[cos_{\text{demod}}\ (t)]|_{Bessel\,,1kHz} \tag{8}$$

$$sin_{\text{demod2}}\ (t) = LPF[sin_{\text{demod}}\ (t)]|_{Bessel\,,1kHz} \tag{9}$$

Obviously, the applied LPF introduces a phase delay, as can be seen in Fig. 14.

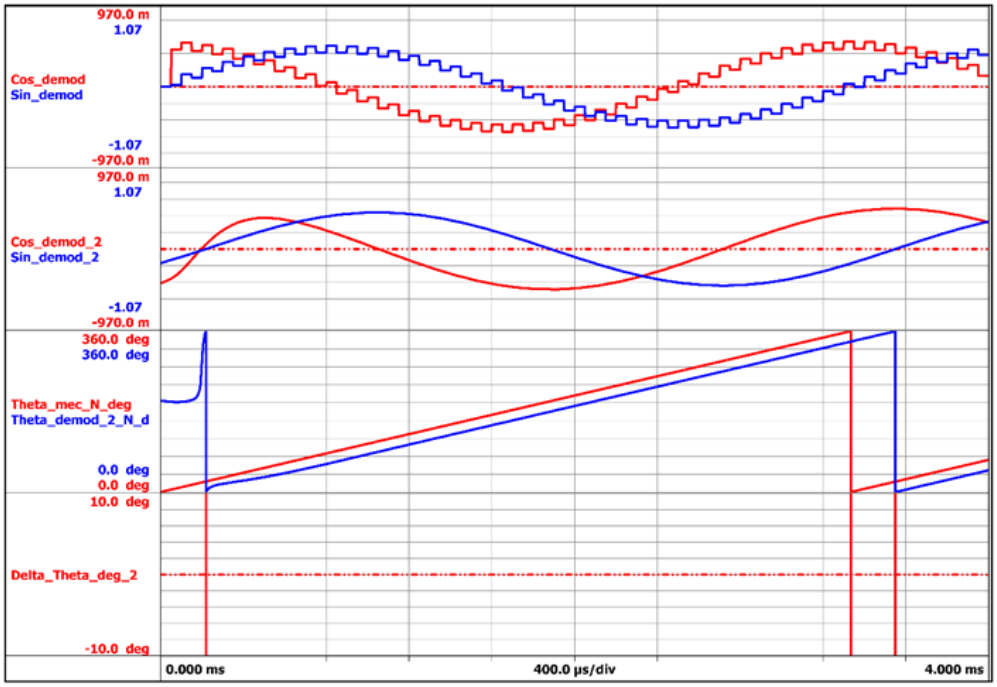

Fig. 14: Simulated steady-state operation at 18000 rpm with offset scenario 2. From top to bottom: $cos_{demod,}$, sin_{demod}, filtered $cos_{demod2,}$, sin_{demod2}, real rotor position (red) and new position (blue) obtained from the filtered sine and cosine.

The phase delay can be compensated using an approximated formula as

$$\vartheta_{delay} = \left(\frac{1}{\omega_{LPF}} + 0.5\frac{1}{f_{ecc}} + \tau\right) \cdot \omega_{mec} \tag{10}$$

where ω_{LPF} (rad/s) is the LPF cut-off frequency, ω_{mec} (rad/s) is the mechanical speed that is computed as the mean value over one cycle defined by the reconstructed angle, τ is an additional correction time of 6 µs.

The correction proposed by (10) is effective, as shown in Fig. 15; the error between the real position and the reconstructed position has some oscillations having very low amplitude (0.2 degrees).

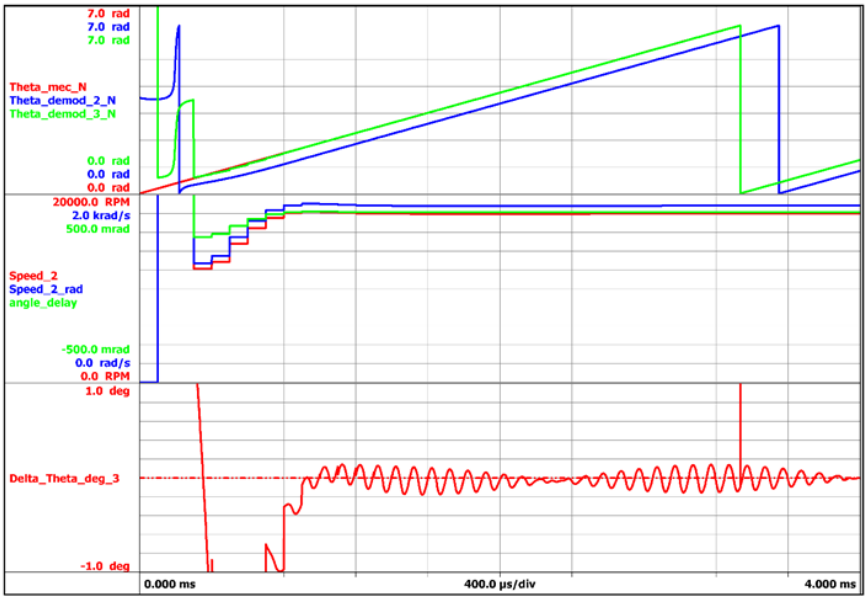

Fig. 15: Simulated steady-state operation at 18000 rpm with offset scenario 2. From top to bottom: real rotor position (red), reconstructed rotor position without phase correction (blue) new position (green) with phase correction, measured speed (rad/s and rpm) and angle delay (green), error between the real position and new position obtained with phase correction.

3. Experimental Results

The above described method has been tested for an LTN resolver using the GEN7t data recorder, as shown in Fig. 16.

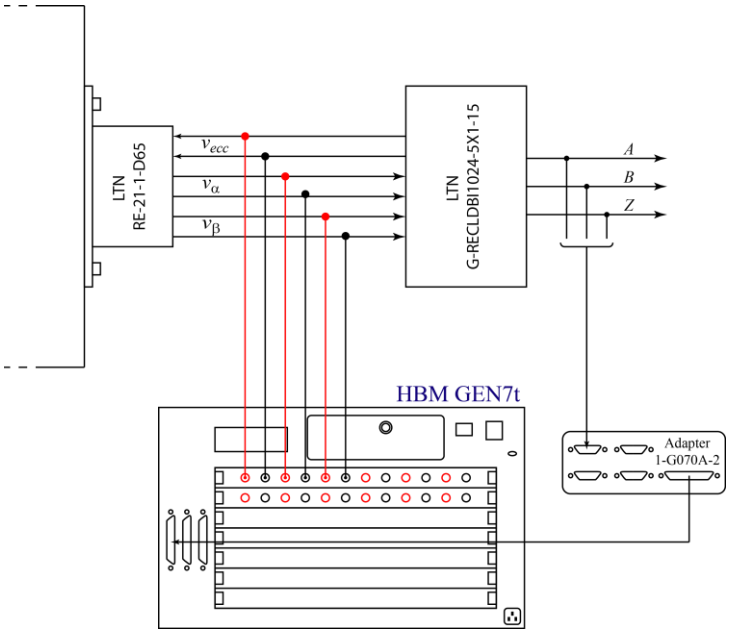

Fig. 16: Resolver testing layout using GEN7t data recorder.

The excitation voltage v_{ecc} and the resolver outputs $v_{\alpha\beta}$ are acquired by the data recorder using three voltage channels of a fast acquisition board GN610B (sample rate 2MS/s @18 bit per channel, accuracy better than 0.1%). The formulas used by the Perception workbench are provided in the Appendix. The resolver signals have been applied to a resolver-to-encoder conversion board that generates three encoder signals (A; B and Z), which have been acquired by the data recorder using one digital port. The encoder signals have been used to obtain a position used as reference for the evaluation of the angle reconstruction method.

The experiment has been performed at low speed and the results are shown in Figs. 17-19.

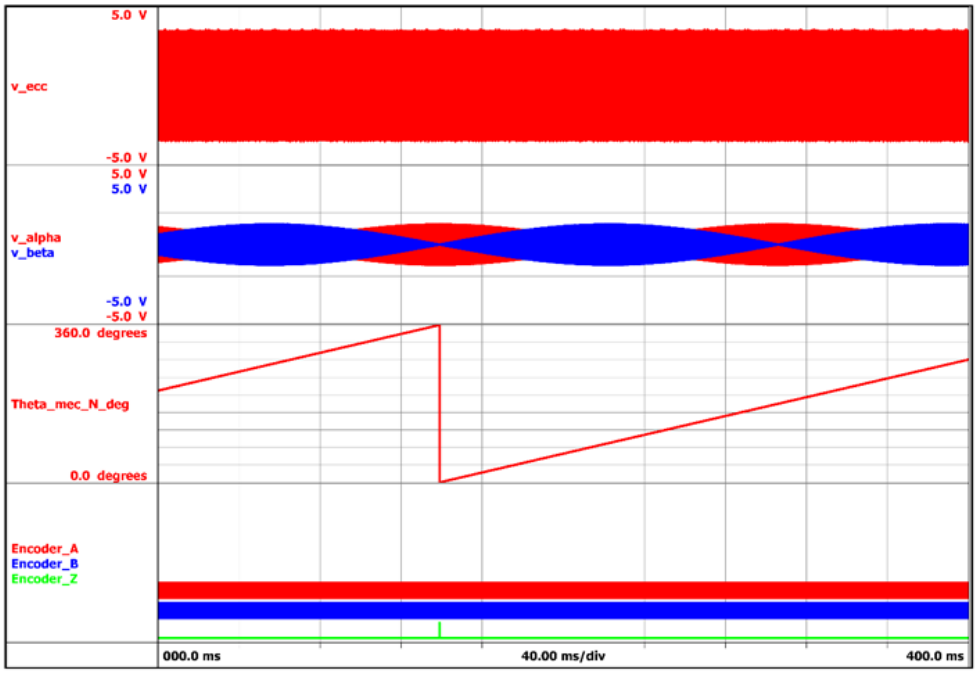

Fig. 17: Experimental result. From top to bottom: excitation voltage (red), v_α (red), v_β (blue), measured position from encoder signals, encoder signals A,B and Z.

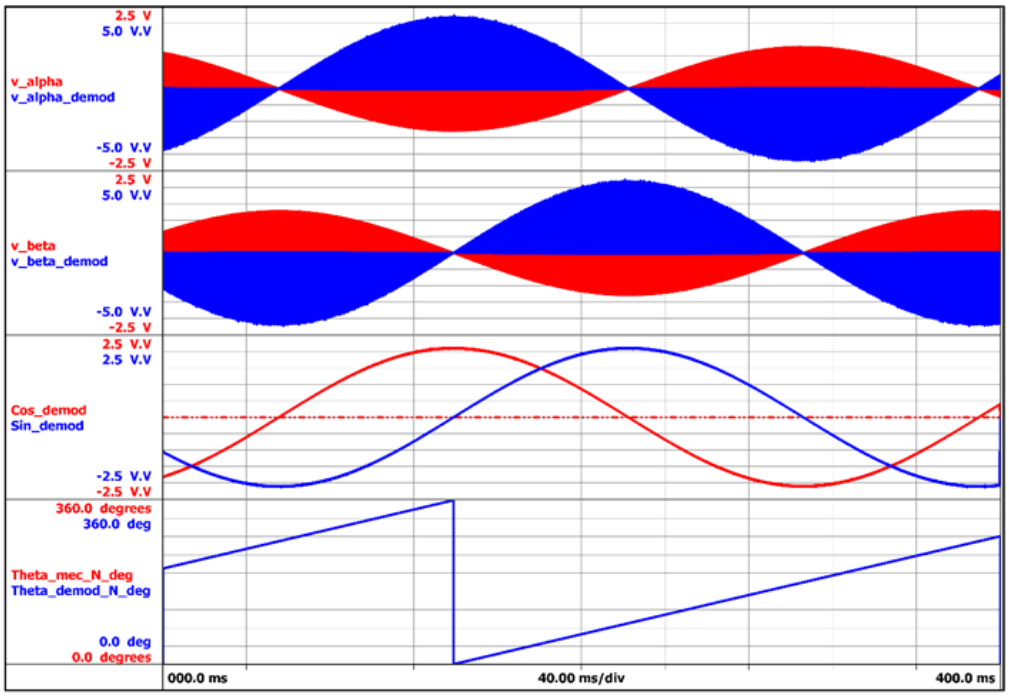

Fig. 18: Experimental result. From top to bottom: v_α (red), $v_{\alpha,demod}$ (blue), v_β (red), $v_{\beta,demod}$ (blue), demodulated sine and cosine $cos_{demod,,}$ sin_{demod}, encoder position (red) and reconstructed position (blue).

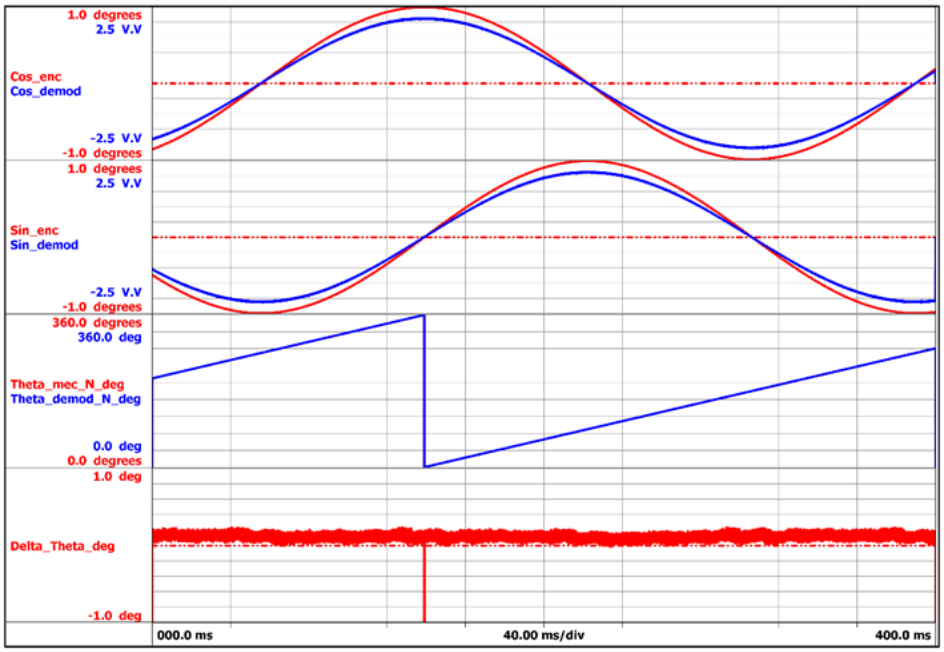

Fig. 19: Experimental result. From top to bottom: cosine of encoder position (red), demodulated cosine cos_{demod} (blue), sine of encoder position (red), demodulated sine sin_{demod} (blue), encoder position (red) and reconstructed position (blue), error between encoder position and reconstructed position.

A zoom view of Fig. 19 is shown in Fig. 20, wherein the error between the encoder position and the reconstructed position is clearly discernible. This error has a non-zero mean value that is probably produced by the delay introduced by the board (see Fig. 16), which elaborates the encoder signals from the resolver outputs.

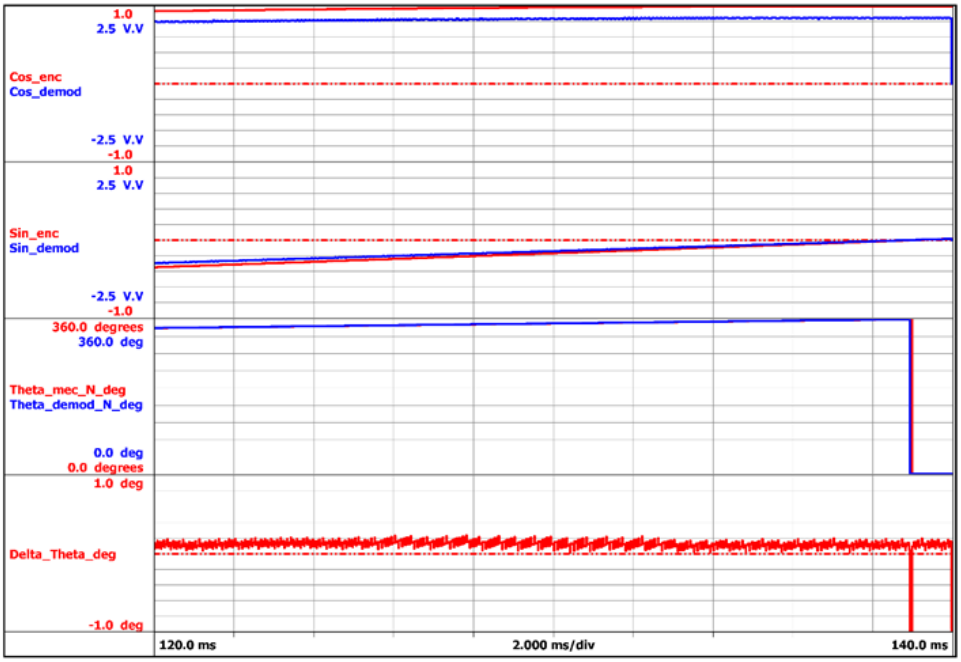

Fig. 20: Experimental results. Zoom view from Fig. 19. From top to bottom: cosine of encoder position (red), demodulated cosine cos_{demod} (blue), sine of encoder position (red), demodulated sine sin_{demod} (blue), encoder position (red) and reconstructed position (blue), error between encoder position and reconstructed position.

Low pass filters can be applied to the demodulated sine and cosine $\cos_{demod}$, $\sin_{demod}$ to compensate the offsets and high frequency oscillations. Fig. 21 shows the experimental results that are obtained with filter and phase correction.

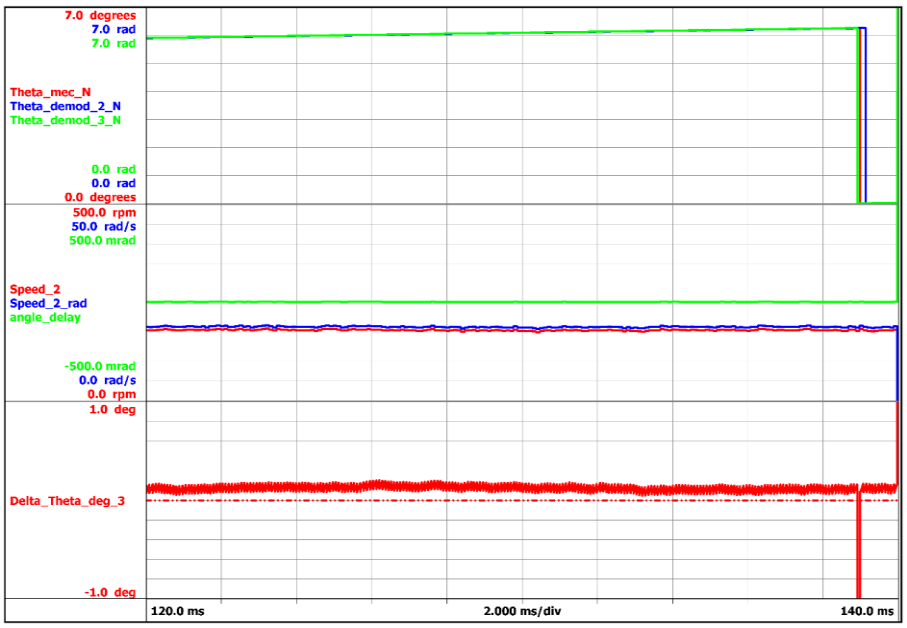

Fig. 21: Experimental result- From top to bottom: encoder position (red), reconstructed rotor position without phase correction (blue) new position with phase correction (green), measured speed (rad/s and rpm) and angle delay (green), error between the encoder position and new position obtained with phase correction.

A zoom view of the position errors is shown in Fig. 22. This result is extremely interesting, since the error that is obtained with the filtering completely eliminates the jumps that are produced by the cycle detection of the excitation voltage v_{ecc}. The sawtooth waveform of the error that is obtained after the filtering is exclusively produced by the effects of the encoder quantization.

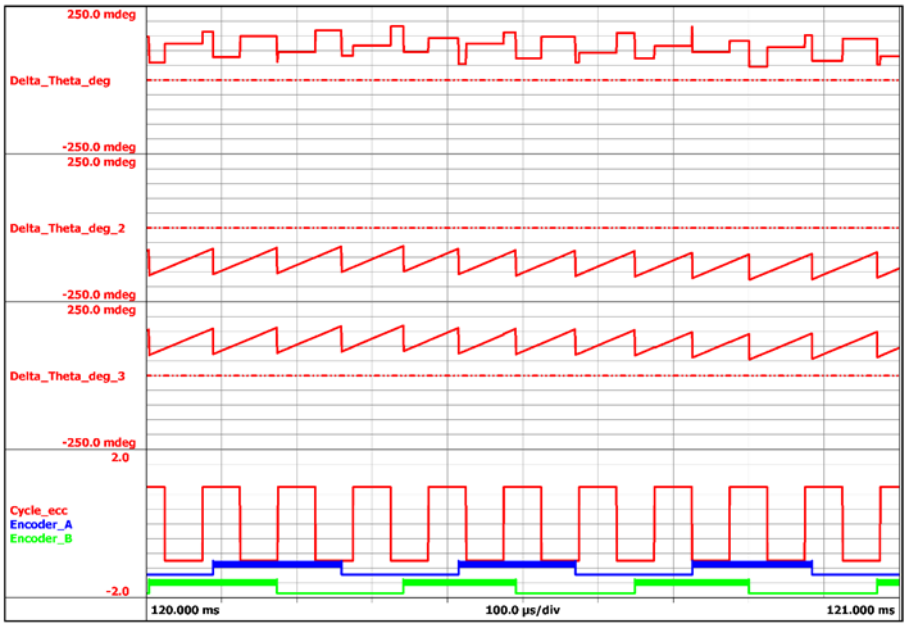

Fig. 22: Experimental result - From top to bottom: error between the encoder position and reconstructed position, error between the encoder position and new position without phase correction, error between the encoder position and new position with phase correction, cycle detection of the excitation signal and the encoder channels A, B.

4. Conclusions

This work presented a method to measure the position provided by a resolver by means of a proper demodulation method. The resolver output signals are acquired with a high speed acquisition board. The effects of offsets can be eliminated with low pass filters applied to the demodulated sine and cosine signals, and the phase delay can be compensated with a proper phase correction that depends on the operating speed.

Acknowledgment

The authors are grateful to HBM Italia Srl and HBM Deutschland for the support provided in the data acquisition system realization.

Appendix

The formulas provided below are divided into formulas for resolver simulation and signal elaboration, and formulas for the elaboration of the acquired resolver signals.

(A) Formulas for resolver simulation

1		Resolver simulation	
2		#region	
3		Simulation parameter	
4		Simulated time, Start time, Stop time	
5	param_time_start	0	s
6	param_time_end	0.1	s
7		Sampling frequency and number of samples	
8	param_f_sample	2000000	S/s
9	param_Nsample	(Formula.param_time_end-Formula.param_time_start)* Formula.param_f_sample	
10		**********	
11		Simulated Resolver Parameter	
12		Excitation frequency for the resolver	
13	param_f_ecc	10000	Hz
14		Coupling factor of the output coil	
15	param_k_alpha	1	
16	param_k_beta	1	
17		Sampling DC offset on the output signal of the resolver	
18	param_offset_alpha	0	
19	param_offset_beta	0	
20		Mechanical speed and mechanical frequency	
21	param_speed	3000	Rpm
22	param_Freq_mec	Formula.param_speed/60	Hz
23		Filter parameter	

24	param_Filter_frequency	1000	Hz
25	param_Filter_order	2	
26	param_Filter_phaseless	0	
27		Constants	
28	const_Deg2Rad	System.Constants.Pi/180	
29	const_Rad2Deg	180/System.Constants.Pi	
30	const_RPM2Hz	1/60	
31	const_Hz2Rad_s	2*System.Constants.Pi	
32	const_RPM2rad	System.Constants.Pi/30	
33		Number of cycles for the demodulation	
34	param_Cycle_Exct	0.5	
35		Number of cycles for speed computation	
36	param_Cycle_speed	1	
37		#endregion	
38		**********	
39		Mechanical quantities generation and resolver simulation	
40		#region	
41	Sin_Mech	@SineWave(Formula.param_f_sample; Formula.param_Nsample; Formula.param_Freq_mec; 0)	
42	Cos_Mech	@SineWave(Formula.param_f_sample; Formula.param_Nsample; Formula.param_Freq_mec; 90)	
43	Theta_mec	@SpaceVectorInverseTransformation(Formula.Cos_Mech; Formula.Sin_Mech; 0; 4; 0)	
44		Normalized angle	
45	Theta_mec_N	@Modulo((Formula.Theta_mec+2*System.Constants.Pi);2*System.Constants.Pi)	rad
46	Theta_mec_N_deg	Formula.Theta_mec_N*Formula.const_Rad2Deg	deg
47		**********	
48		Resolver simulation	
49		Excitation voltage of the Resolver	
50	v_ecc	@SineWave(Formula.param_f_sample; Formula.param_Nsample; Formula.param_f_ecc; 0)	
51		Output of the simulated Resolver	
52	v_alpha	Formula.Cos_Mech*Formula.v_ecc*Formula.param_k_alpha+ Formula.param_offset_alpha	
53	v_beta	Formula.Sin_Mech*Formula.v_ecc*Formula.param_k_beta+ Formula.param_offset_beta	
54		#endregion	
55		**********	

56		Demodulation formulas	
57		#region	
58		Resolver output demodulated voltage	
59	v_alpha_demod	Formula.v_alpha*Formula.v_ecc	
60	v_beta_demod	Formula.v_beta*Formula.v_ecc	
61		Cycle definition from Excitation	
62	Cycle_ecc	@CycleDetect(Formula.v_ecc; 0; 0.1)	
63		output mechanical Sin and Cos	
64	Cos_demod	@CycleMean(Formula.v_alpha_demod; Formula.param_Cycle_number; Formula.Cycle_ecc)	
65	Sin_demod	@CycleMean(Formula.v_beta_demod; Formula.param_Cycle_number; Formula.Cycle_ecc)	
66		Demodulated mechanical angle	
67	Theta_demod	@SpaceVectorInverseTransformation(Formula.Cos_demod; Formula.Sin_demod; 0; 4; 0)	
68	Theta_demod_N	@Modulo((Formula.Theta_demod+2*System.Constants.Pi);2*System.Constants.Pi)	
69	Theta_demod_N_deg	Formula.Theta_demod_N*Formula.const_Rad2Deg	
70		Error	
71	Delta_Theta_deg	Formula.Theta_demod_N_deg-Formula.Theta_mec_N_deg	deg
72		#endregion	
73		**********	
74		Offset attenuation effect by filter	
75		#region	
76		Filter	
77	Cos_demod_2	@FilterBesselLP(Formula.Cos_demod; Formula.param_Filter_order; Formula.param_Filter_frequency; Formula.param_Filter_phaseless)	
78	Sin_demod_2	@FilterBesselLP(Formula.Sin_demod; Formula.param_Filter_order; Formula.param_Filter_frequency; Formula.param_Filter_phaseless)	
79		demodulated angle	
80	Theta_demod_2	@SpaceVectorInverseTransformation(Formula.Cos_demod_2; Formula.Sin_demod_2; 0; 4; 0)	
81	Theta_demod_2_N	@Modulo((Formula.Theta_demod_2+2*System.Constants.Pi); 2*System.Constants.Pi)	rad
82	Theta_demod_2_N_deg	Formula.Theta_demod_2_N*Formula.const_Rad2Deg	deg
83		Error	
84	Delta_Theta_deg_2	Formula.Theta_demod_2_N_deg-Formula.Theta_mec_N_deg	deg
85		#endregion	
86		**********	

87		Filter delay compensation	
88		#region	
89		Speed computation	
90	Speed_2	@CycleRPM(Formula.Theta_demod_2_N_deg; Formula.param_Cycle_speed; Formula.Cycle_ecc)	RPM
91	Speed_2_rad	Formula.Speed_2*Formula.const_RPM2rad	rad/s
92		delay evaluation	
93	Ecc_frequncy	@CycleFrequency(Formula.v_ecc; 1; Formula.Cycle_ecc)	Hz
94	delay_eq	1/(Formula.param_Filter_frequency*Formula.const_Hz2Rad_s)+ Formula.param_Cycle_number/Formula.Ecc_frequncy+0.000006	s
95	angle_delay	Formula.delay_eq*Formula.Speed_2_rad	rad
96		angle correction	
97	Theta_demod_3_N	@Modulo((Formula.Theta_demod_2_N+Formula.angle_delay); 2*System.Constants.Pi)	rad
98	Theta_demod_3_N_deg	Formula.Theta_demod_3_N*Formula.const_Rad2Deg	deg
99		Error	
100	Delta_Theta_deg_3	Formula.Theta_demod_3_N_deg-Formula.Theta_mec_N_deg	deg
101		#endregion	
102			

(B) Formulas for resolver elaboration for the experimental results

The formulas are divided into 5 regions as follows:

- The parameters and constant values are defined from line 2 to line 20.
- The assignment of acquired variables is performed from line 23 to line 32.
- The demodulation and angle reconstruction formulas are implemented from line 35 up to line 53.
- The filters are implemented from line 57 up to line 66.
- The phase correction is implemented from line 69 up to line 82.

1		Elaboration Parameter and Constant	
2		#region	
3		Time limitation	
4	Time_Start	0	s
5	Time_Stop	0.4	s
6		Filter parameter	
7	param_Filter_frequency	1000	Hz
8	param_Filter_order	2	
9	param_Filter_phaseless	0	

10		Constants	
11	const_Deg2Rad	System.Constants.Pi/180	
12	const_Rad2Deg	180/System.Constants.Pi	
13	const_RPM2Hz	1/60	
14	const_Hz2Rad_s	2*System.Constants.Pi	
15	const_RPM2rad	System.Constants.Pi/30	
16		Demodulation period	
17	param_Cycle_number	0.5	
18		Number of cycles for speed computation	
19	param_Cycle_speed	1	
20		#endregion	
21		**********	
22		Input assignment	
23		#region	
24	v_alpha	@Cut(Active.Group1.Recorder_A.Cos; Formula.Time_Start; Formula.Time_Stop)	
25	v_beta	@Cut(Active.Group1.Recorder_A.Sin; Formula.Time_Start; Formula.Time_Stop)	
26	v_ecc	@Cut(Active.Group1.Recorder_A.V_ref; Formula.Time_Start; Formula.Time_Stop)	
27	Theta_mec_N_deg	@Cut(Active.Group1.Recorder_B.Angle_mec; Formula.Time_Start; Formula.Time_Stop)	
28	Encoder_A	@Cut(Active.Group1.Recorder_B.Ev_B7_02; Formula.Time_Start; Formula.Time_Stop)	
29	Encoder_B	@Cut(Active.Group1.Recorder_B.Ev_B7_03; Formula.Time_Start; Formula.Time_Stop)	
30	Encoder_Z	@Cut(Active.Group1.Recorder_B.Ev_B7_01; Formula.Time_Start; Formula.Time_Stop)	
31	Theta_mec_N	Formula.Theta_mec_N_deg*Formula.const_Deg2Rad	
32		#endregion	
33		**********	
34		Demodulation formulas	
35		#region	
36		Resolver output demodulated voltage	
37	v_alpha_demod	Formula.v_alpha*Formula.v_ecc	
38	v_beta_demod	Formula.v_beta*Formula.v_ecc	
39		Cycle definition from Excitation	
40	Cycle_ecc	@CycleDetect(Formula.v_ecc; 0; 0.1)	
41		output mechanical sin and cos	
42	Cos_demod	@CycleMean(Formula.v_alpha_demod;	

		Formula.param_Cycle_number; Formula.Cycle_ecc)	
43	Sin_demod	@CycleMean(Formula.v_beta_demod; Formula.param_Cycle_number; Formula.Cycle_ecc)	
44		Demodulated mechanical angle	
45	Theta_demod	@SpaceVectorInverseTransformation(Formula.Cos_demod; Formula.Sin_demod; 0; 4; 0)	
46	Theta_demod_N	@Modulo((Formula.Theta_demod+2*System.Constants.Pi);2*System.Constants.Pi)	
47	Theta_demod_N_deg	Formula.Theta_demod_N*Formula.const_Rad2Deg	deg
48		Simulated encoder Sin and Cos	
49	Cos_enc	@Cosine(Formula.Theta_mec_N)	
50	Sin_enc	@Sine(Formula.Theta_mec_N)	
51		Error	
52	Delta_Theta_deg	Formula.Theta_demod_N_deg-Formula.Theta_mec_N_deg	
53		#endregion	
54		**********	
55		Offset attenuetion effect by filter	
56		#region	
57		Filter	
58	Cos_demod_2	@FilterBesselLP(Formula.Cos_demod; Formula.param_Filter_order; Formula.param_Filter_frequency; Formula.param_Filter_phaseless)	
59	Sin_demod_2	@FilterBesselLP(Formula.Sin_demod; Formula.param_Filter_order; Formula.param_Filter_frequency; Formula.param_Filter_phaseless)	
60		demodulated angle	
61	Theta_demod_2	@SpaceVectorInverseTransformation(Formula.Cos_demod_2; Formula.Sin_demod_2; 0; 4; 0)	
62	Theta_demod_2_N	@Modulo((Formula.Theta_demod_2+2*System.Constants.Pi); 2*System.Constants.Pi)	
63	Theta_demod_2_N_deg	Formula.Theta_demod_2_N*Formula.const_Rad2Deg	deg
64		Error	
65	Delta_Theta_deg_2	Formula.Theta_demod_2_N_deg-Formula.Theta_mec_N_deg	
66		#endregion	
67		**********	
68		Filter delay compensation	
69		#region	
70		Speed computation	
71	Speed_2	@CycleRPM(Formula.Theta_demod_2_N_deg; Formula.param_Cycle_speed; Formula.Cycle_ecc)	
72	Speed_2_rad	Formula.Speed_2*Formula.const_RPM2rad	

73		delay evaluation	
74	Ecc_frequncy	@CycleFrequency(Formula.v_ecc; 1; Formula.Cycle_ecc)	
75	delay_eq	1/(Formula.param_Filter_frequency*Formula.const_Hz2Rad_s)+ Formula.param_Cycle_number/Formula.Ecc_frequncy+0.000007	
76	angle_delay	Formula.delay_eq*Formula.Speed_2_rad	
77		angle correction	
78	Theta_demod_3_N	@Modulo((Formula.Theta_demod_2_N+Formula.angle_delay);2*System.Constants.Pi)	
79	Theta_demod_3_N_deg	Formula.Theta_demod_3_N*Formula.const_Rad2Deg	
80		Error	
81	Delta_Theta_deg_3	Formula.Theta_demod_3_N_deg-Formula.Theta_mec_N_deg	deg
82		#endregion	