AN4958
Application note

L9369 ADC accuracy

Alessandro Vaghi

Introduction

Electronic Parking Brakes (EPB) devices provide 10 Analog to Digital Converter (ADC) channels to get bridges' voltages and current measurements.

Purpose of this document is to describe the accuracy of ADC channels and to show some strategy to compensate possible errors.

Contents

1 ADC channels

The following channels are available:

Table 1. ADC channels

CH #	CH Name	Description
1	VSBRIDGE_A	Drain connection of the external H-Bridge A High-side NFET and measurement input for H-bridge supply voltage
2	VSBRIDGE_B	Drain connection of the external H-Bridge B High-side NFET and measurement input for H-bridge supply voltage
3	SH1_A	Side 1 source connection of the H-Bridge High-side NFET for wheel brake actuator A
4	SH1_B	Side 1 source connection of the H-Bridge High-side NFET for wheel brake actuator B
5	SH2_A	Side 2 source connection of the H-Bridge High-side NFET for wheel brake actuator A
6	SH2_B	Side 2 source connection of the H-Bridge High-side NFET for wheel brake actuator B
7	CS1_A	Motor current high side sense of external H-Bridge A
8	CS1_B	Motor current high side sense of external H-Bridge B
9	CS2_A	Motor current low side sense of external H-Bridge A
10	CS2_B	Motor current low side sense of external H-Bridge B

ADC data values can be read through SPI registers together with synchronization and H-Bridge status information.

2 ADC channels data

ADC 12 and 13 bit data need to be converted to obtain the corresponding voltage and current values; the following formulas have to be applied:

1) The channel code is inverted

Table 2. ADC data conversion

CH #	CH Name	Conversion formulas (CODE is the ADC bare data)	Bit n.	LSB	Range
1	VSBRIDGE_A	VSB B (V) = 34.2 * (4095 - CODE) / 4096	12 inv.[1]	8.35 mV	0÷34.2 V
2	VSBRIDGE_B	VSB-A (V) = 34.2 * CODE / 4096	12	8.35 mV	0÷34.2 V
3	SH1_A	VSH1A (V) = 34.2 * (CODE – 4096) / 4096	13	8.35 mV	±34.2 V
4	SH1_B	VSH1B (V) = 34.2 * (CODE – 4096) / 4096	13	8.35 mV	±34.2 V
5	SH2_A	VSH2A (V) = 34.2 * ((8191 - CODE) – 4096) / 4096	13 inv.[1]	8.35 mV	±34.2 V
6	SH2_B	VSH2B (V) = 34.2 * ((8191 - CODE) – 4096) / 4096	13 inv.[1]	8.35 mV	±34.2 V
7	CS1_A	I(A) = ((CODE/8192 x 410mV) - 205 mV) / Rsense[2]	13	17/25 mA[3]	±68/102 A
8	CS1_B	I(A) = ((CODE/8192 x 410mV) - 205 mV) / Rsense[2]	13	17/25 mA[3]	±68/102 A
9	CS2_A	I(A) = ((CODE/8192 x 410mV) - 205 mV) / Rsense[2]	13	17/25 mA[3]	±68/102 A
10	CS2_B	I(A) = ((CODE/8192 x 410mV) - 205 mV) / Rsense[2]	13	17/25 mA[3]	±68/102 A

1. The channel code is inverted.
2. R_{sense} expressed in mΩ.
3. The LSB voltage weight is ~50 µA corresponding to ~25mA for a 2 mΩ shunt resistor and to ~17mA for a 3 mΩ shunt.

3 ADC accuracy

ADC channels can be divided into three different categories:

- 12 bit voltage measurement channels: VSBRIDGE_A and VSBRIDGE_B;
- 13 bit voltage measurement channels: SH1_A, SH2_A, SH1_B and SH2_B;
- 13 bit current measurement channels: CSA1_A, CSA2_A, CSA1_B and CSA2_B.

3.1 VSBRIDGE accuracy

Table 3. VSBRIDGE accuracy

Voltage range	Accuracy
VSBRIDGEx ≤ 5 V	± 200 mV max
VSBRIDGEx > 5 V	± 4% max

3.2 SH accuracy

Table 4. SH accuracy

Voltage range	Accuracy
SHx_y ≤ 4.8 V	± 216 mV max
SHx_y > 4.8 V	± 4.5% max

The accuracy in measuring the voltage difference between SH1_y and SH2_y pairs (matching) is 6% max.

3.3 CSA accuracy

Current sense data are affected by an offset error at 0A: a correct accuracy measurement requires an offset compensation.

Because of the current measurement redundancy data of CSA1_y and CSA2_y channel can be used to increase the overall current measurement accuracy.

3.3.1 CSA offset at 0A

The maximum current offset a **I** = 0 A is ±16.8LSB corresponding to ~280 mA for a 3 mΩ shunt resistor and ~420 mA for a 2 mΩ shunt resistor.

3.3.2 CSA single channel offset compensation

The offset compensation is performed subtracting the offset from the ADC channel data; using this strategy the single current sense channel accuracy, in terms of voltage, is:

Table 5. Single CSA channel accuracy after offset compensation

Voltage absolute value range	Accuracy
0 mV < |V| < 2.4 mV	±14LSB
2.4 mV ≤ |V| < 7.5 mV	±9.75LSB
7.5 mV ≤ |V| ≤ 205 mV	±6.5%

3.3.3 CSA average compensation

CSA1_y and CSA2_y channels measure the voltage across two shunt resistors crossed by the same current flow; the following accuracy can be obtained computing the average of the two channels:

Table 6. Single CSA channel accuracy after offset compensation and average

Voltage absolute value range	Accuracy
2.4 m V < |V| ≤ 3.0 mV	±16.5%
3.0 mV < |V| ≤ 6.0 mV	±11.6%
6.0 mV < |V| ≤ 9.0 mV	±6.7%
9.0 mV < |V| ≤ 205 mV	±4.8%

3.3.4 C code example

The following example shows a very simplified C code to compute the offset compensation and the average:

```
#define CSA_0_CURRENT_CODE          0x1000

s16 CSA1_A_Offset = 0 ; // Signed 16 bit
s16 CSA2_A_Offset = 0 ;
s16 CSA1_B_Offset = 0 ;
s16 CSA2_B_Offset = 0 ;

/***************************************************************************
*****/
/*                                                                        */
/* void SetOffset ( s16 CSA1_A_0_CurrentCode , s16 CSA2_A_0_CurrentCode ,
*/
/*                 s16 CSA1_B_0_CurrentCode , s16 CSA2_B_0_CurrentCode )
*/
/*                                                                        */
/***************************************************************************
*****/

void SetOffset ( s16 CSA1_A_0_CurrentCode , s16 CSA2_A_0_CurrentCode ,
                 s16 CSA1_B_0_CurrentCode , s16 CSA2_B_0_CurrentCode )
{
  /* Signed subtractions between ADC code read at 0 current */
  /* condition and the expected value                        */

  CSA1_A_Offset = CSA1_A_0_CurrentCode - CSA_0_CURRENT_CODE ;
  CSA2_A_Offset = CSA2_A_0_CurrentCode - CSA_0_CURRENT_CODE ;
  CSA1_B_Offset = CSA1_B_0_CurrentCode - CSA_0_CURRENT_CODE ;
  CSA2_B_Offset = CSA2_B_0_CurrentCode - CSA_0_CURRENT_CODE ;

} /* end of SetOffset() function */

/***************************************************************************
*****/
/*                                                                        */
/* void GetCSAValues ( s16 CSA1_A_Code , s16 CSA2_A_Code ,
*/
/*                     s16 CSA1_B_Code , s16 CSA2_B_Code )
*/
/*                     s16 *CSA_A_Val  , s16 *CSA_B_Val )
*/
/*                                                                        */
/***************************************************************************
*****/
```

```
void GetCSAValues ( s16 CSA1_A_Code , s16 CSA2_A_Code ,
                    s16 CSA1_B_Code , s16 CSA2_B_Code )
                    s16 *CSA_A_Val  , s16 *CSA_B_Val )
{
  /* Subtract the offset from each ADC value and compute the average */
  /* of the two currents measured on the same H-Bridge              */

  *CSA_A_Val = ( ( CSA1_A_Code - CSA1_A_Offset ) + ( CSA2_A_Code -
CSA2_A_Offset ) ) / 2 ;
  *CSA_B_Val = ( ( CSA1_B_Code - CSA1_B_Offset ) + ( CSA2_B_Code -
CSA2_B_Offset ) ) / 2 ;

} /* end of GetCSAValues () function */
```

The offset computation could be executed on system startup and/or periodically and/or just before H-Bridges' actuation.

3.4 ADC formulas with offset

To take into account 0V or 0A offsets ADC formulas of table 2 are still valid but the CODE variable has to be substituted with (CODE - OFFSET) where OFFSET is the difference between the expected value at 0V or 0A and the effectively read code.

3.4.1 VS_Bridge_A

If $CODE_{0V}$ is the code returned by the ADC at 0V, the offset is:

OFFSET = 4095 - $CODE_{0V}$.

The VS_Bridge_A computing formula becomes:

Equation 1: $$VSB_A(V) = 34.2 \times \frac{((4095 - CODE) - OFFSET)}{4096}$$

3.4.2 VS_Bridge_B

If $CODE_{0V}$ is the code returned by the ADC at 0V, the offset is:

OFFSET = $CODE_{0V}$.

The VS_Bridge_B computing formula becomes:

Equation 2: $$VSB_B(V) = 34.2 \times \frac{(CODE - OFFSET)}{4096}$$

3.4.3 SH1_A and SH1_B

If $CODE_{0V}$ is the code returned by the ADC at 0V, the offset is:

OFFSET = $CODE_{0V}$ - 4096.

The SH1_A and SH1_B computing formula becomes:

Equation 3: $$SH1_{A/B}(V) = 34.2 \times \frac{(CODE - 4096 - OFFSET)}{4096}$$

3.4.4 SH2_A and SH2_B

If $CODE_{0V}$ is the code returned by the ADC at 0V, the offset is:

OFFSET = $CODE_{0V}$ - 4096.

The SH2_A and SH2_B computing formula becomes:

Equation 4: $$SH2_{A/B}(V) = 34.2 \times \frac{((8191 - CODE) - OFFSET - 4096)}{4096}$$

3.4.5 CSA

If $CODE_{0V}$ is the code returned by the ADC at 0A (and 0V), the offset is:

OFFSET = $CODE_{0V}$ - 4096.

The CSA computing formula becomes:

Equation 5: $$V_{CSA}(mV) = \left(410 \times \frac{(CODE - OFFSET)}{8192}\right) - 205$$

Equation 6: $$I_{CSA}(A) = \frac{V_{CSA}\ (mV)}{R_{shunt(mOhm)}} = \frac{\left(410 \times \frac{(CODE - OFFSET)}{8192} - 205\right)}{R_{shunt(mOhm)}}$$

Appendix A CSA correction curves

In the following paragraph CSA curves will be exposed in order to see the correction algorithm behavior.

The graphs show the following data:

Table 7. Graphs' Legenda

Voltage absolute value range	Accuracy
X-axis	Expected voltage
CSA1	Voltage measured at CSA1 pin
CSA1_COMP	Voltage measured at CSA1 pin compensated subtracting 0 V offset
CSA2	Voltage measured at CSA2 pin
CSA2_COMP	Voltage measured at CSA2 pin compensated subtracting 0 V offset
Average	Average value of CSA1_COMP and CSA2_COMP

Figure 1. -3 to 3 mV range

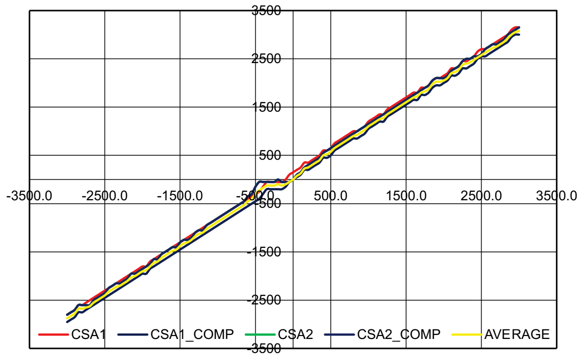


Figure 2. -1 to 1 mV range

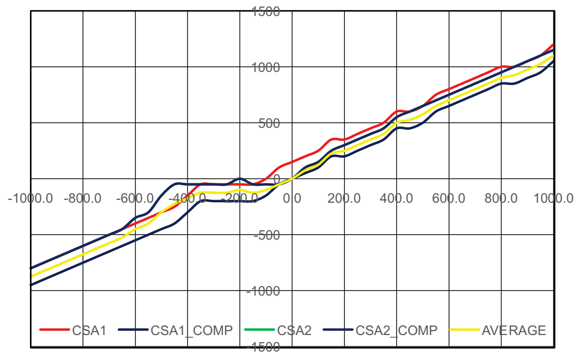


Figure 3. -6 to -3 mV range

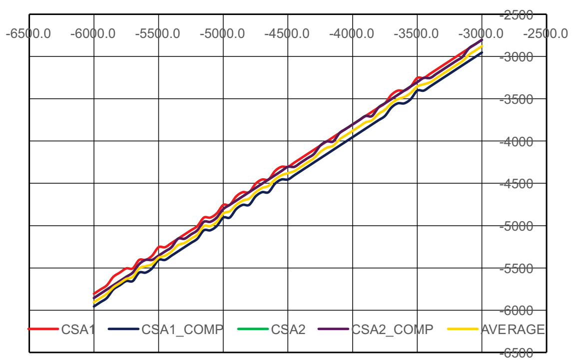


Figure 4. 3 to 6 mV range

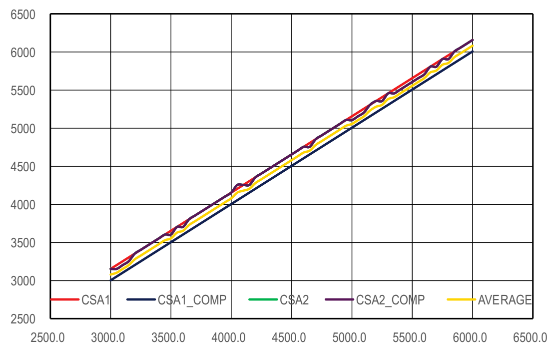


Figure 5. -9 to 6 mV range

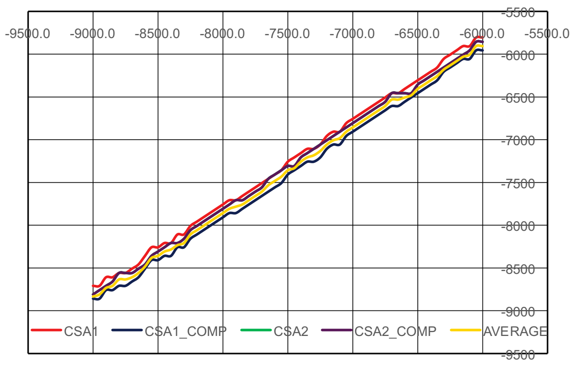


Figure 6. 6 to 9 mV range

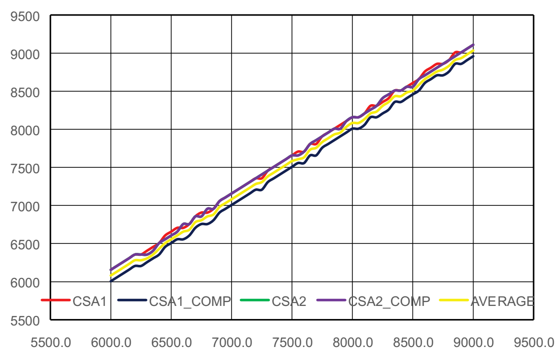


Figure 7. -30 to -9 mV range

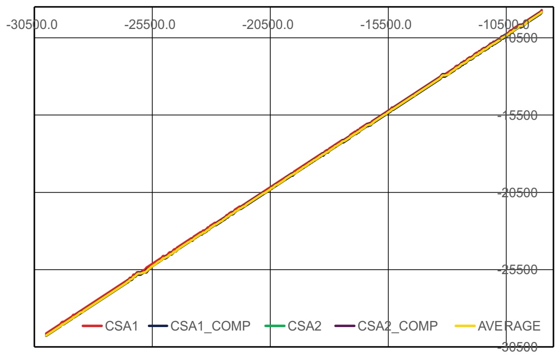
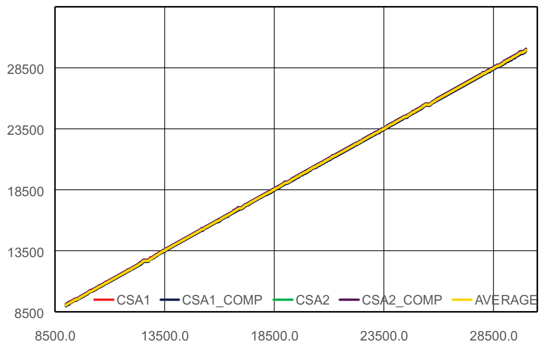


Figure 8. 9 to 30 mV range



Revision history

Table 8. Document revision history

Date	Revision	Changes
05-Jun-2019	1	Initial release.