TWO-POINT METHOD CALIBRATION

Ing. Rafael Oliva

1. Introduction

The processing of data obtained from Analog to Digital (A/D) Converter systems depends very much on the application, but inevitably requires some form of manual or automatic calibration. Calibration of the analog front-end via the “Two Point Method” represents a convenient possibility to minimize the average reading errors of the system. We assume here a typical application of an A/D converter with n bits, and also that the connected computer can use floating point numbers. Furthermore we assume a structure for the conversion of data [Ref.1] as shown in **Figure1**. The front-end of the circuit or “entrance” includes a sensor with a certain linear conversion ratio G_T (V/U.I.), an amplifier with gain A_V, a bias voltage generator V_B, a filter, an ideal multiplexer and an A/D converter with n bits and a reference voltage FSV. The expression U.I. will be used for the magnitude in physical units (e.g.: temperature in ºC or pressure in hPa, etc.)

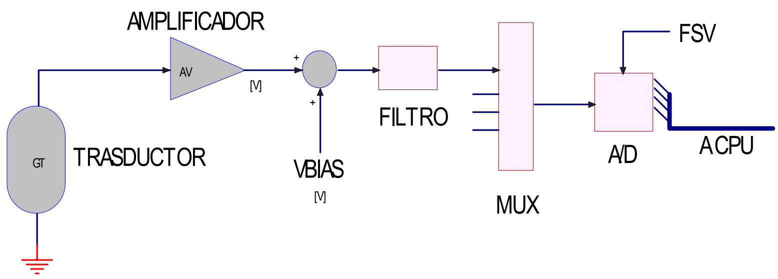


Figure 1 – Diagram of a typical A/D Conversion

An expression for obtaining the physical magnitude U.I. measured, from the A/D converter digital output ADC_{counts} can be written [Ref.1] as:

$$U.I. = \left(\frac{ADC_{counts} * FSV}{2^n - 1} - V_{bias} \right) * \frac{1}{G_T * A_V} \qquad \text{(eq.0)}$$

2. Method implementation

Taking (eq.0), the default values can be conveniently grouped into coefficients C_{off} y G_{conv}, and two new correction coefficients can be defined for calibration purposes: a) the count correction coefficient C_{cal}, with an initial value of 0, and b) the gain correction coefficient G_{cal}, with an initial value of 1.0 , as shown in the following equation:

$$U.I. = \left(ADC_{counts} + C_{offset} + C_{cal}\right) * G_{conv} * G_{cal} \qquad \text{(eq.1)}$$

The two-point method with linear sensors involves comparing two measured values of known precision at different points using (eq.1), and then solving the equation for G_{cal} and C_{cal}. Let us assume we have two measurements $U.I._1$, $U.I._2$ and the corresponding calculated values (*counts*) from the A/D converter, C_1 and C_2. Therefore we get:

$$U.I._1 = \left(C_1 + C_{offset} + C_{cal}\right) * G_{conv} * G_{cal} \qquad \text{(eq.2)}$$
$$U.I._2 = \left(C_2 + C_{offset} + C_{cal}\right) * G_{conv} * G_{cal} \qquad \text{(eq.3)}$$

If we now subtract one equation from the other, the value C_{cal} cancels out and it is possible to solve the equation step by step for G_{cal} and C_{cal}. The result is:

$$G_{cal} = \frac{1}{G_{conv}}\left[\frac{U.I._1 - U.I._2}{C_1 - C_2}\right] \qquad \text{(eq.4)}$$

$$C_{cal} = \frac{U.I._1}{G_{conv} * G_{cal}} - C_1 - C_{offset} \qquad \text{(eq.5)}$$

The implementation of this routine is shown in flow–diagram format in Annex I.
Basically, two successive measurements must be taken, and then the values $U.I._1$ and $U.I._2$, (obtained with an instrument of known precision) must be inserted. In every measurement the program stores the counted values C_1 and C_2. If the sequence is correct, we get equivalent results from (eq. 4 and 5).

The flow diagram corresponds to a routine written in C programming language, in which the main program “passes” a pointer to the routine indicated as HPtr. This pointer allows: a) access to the data of the corresponding channel and b) storage of modified data back in non-volatile memory (NVRAM). For a better understanding of the flow diagram, the following table shows the structure used:

```
struct calib {
        UBYTE   ch;                                     /* Number of Channel used internal  */
        char    Name[CAL_NAMELEN];                      /* Name of Canal                    */
        char    Label[CAL_LABELLEN];                    /* Label of canal, modifiable.      */
        char    SensorTyp[CAL_SENSORTYPLEN];            /* Sensor Type, Nº serie, Fabric.   */
        char    ADCRange[CAL_ADCRANGELEN];              /* String range A/D                 */
        UBYTE   ADCRng;                                 /* Parameter range of MAX197,AIOng  */
        char    Units[CAL_UNITSLEN];                    /* String, units de U.I...          */
        BOOLEAN CalY_N;                                 /* Calibration or no...             */
        char    CalDate[CAL_DATELEN];                   /* ...y if finished.                */
        BOOLEAN EnabledY_N;                             /* Can be disabled...               */
        char    EURange[CAL_EURANGELEN];                /* Range in U.I. ..                 */
        FP      C_Def;                                  /* Parameter default Coffset        */
        FP      G_Def;                                  /* Parameter default Gconv          */
        FP      C_Cal;                                  /* Calibr. Source Ccal Labrosse     */
        FP      G_Cal;                                  /* Calibr. Gain Gcal                */
  };
```

Because of ease of implementation, the program (see flowchart) does not follow exactly the sequence of equations 4 and 5: the calculated coefficients lead first to the result of C_{cal}. This result can be obtained with:

a) a simple operation if U.I.1 = 0 (it is very common to set the first value zero):

$$C_{cal} = -\left(C_1 + C_{offset}\right) \qquad \text{(eq.6)}$$

b) the division of the equations (eq. 10, 11) if U.I.1 ≠ 0. This allows for:

$$\frac{U.I._1}{U.I._2} = \frac{\left(C_1 + C_{offset} + C_{cal}\right) * G_{conv} * G_{cal}}{\left(C_2 + C_{offset} + C_{cal}\right) * G_{conv} * G_{cal}} \qquad \text{(eq.7)}$$

$$C_{cal}\left(\frac{U.I._1}{U.I._2}\right) - C_{cal} = C_1 + C_{offset} - \left(\frac{U.I._1}{U.I._2}\right)\left(C_2 + C_{offset}\right) \qquad \text{(eq.8)}$$

from which:

$$C_{cal} = \frac{C_1 + C_{offset} - \left(\frac{U.I._1}{U.I._2}\right)\left(C_2 + C_{offset}\right)}{\left(\frac{U.I._1}{U.I._2} - 1\right)} \qquad \text{(eq.9)}$$

Finally, for both cases a) and b), it follows that:

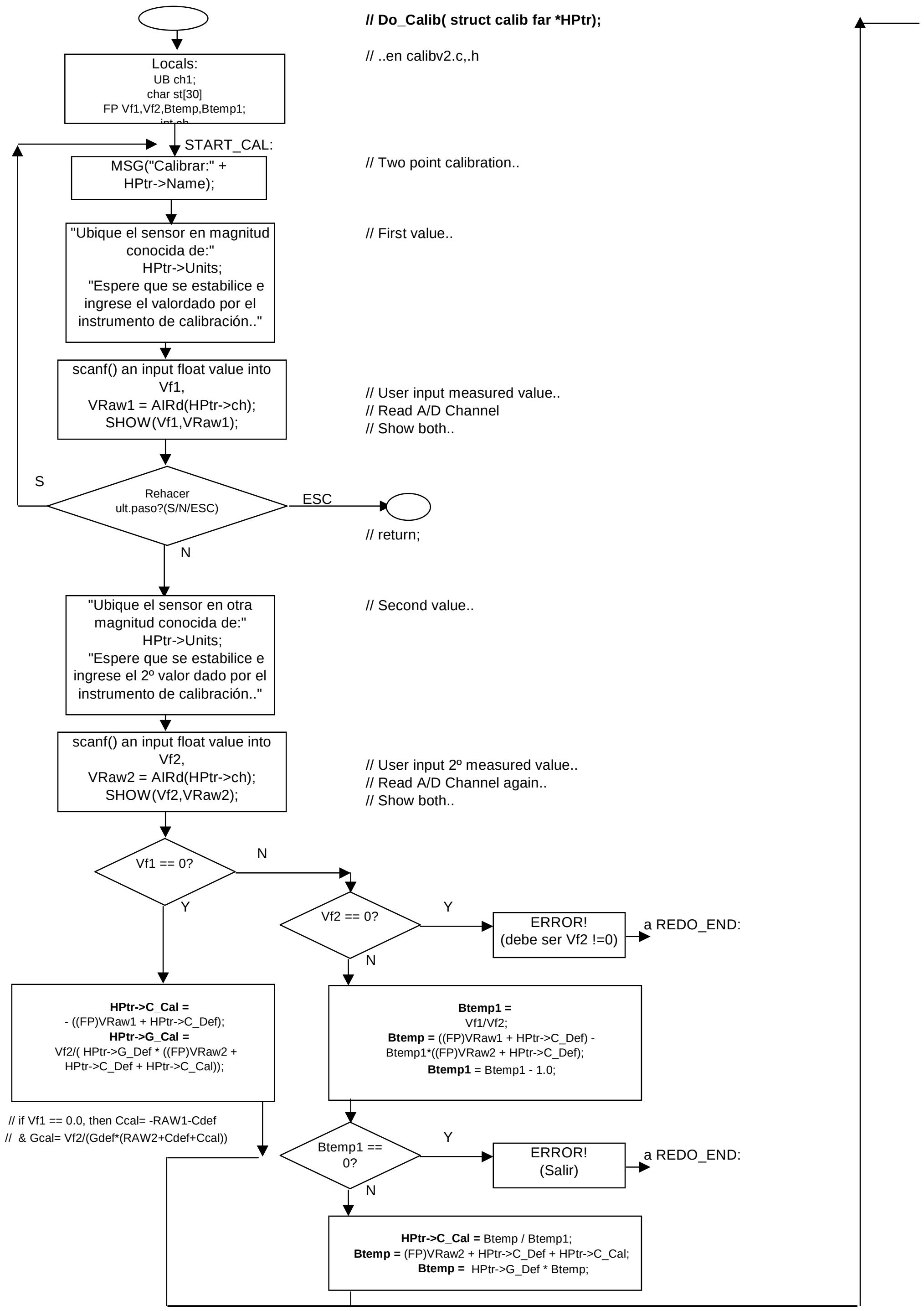
$$G_{cal} = \frac{1}{G_{conv}}\left[\frac{U.I._2}{C_{offset} + C_2 + C_{cal}}\right] \qquad \text{(eq.10)}$$

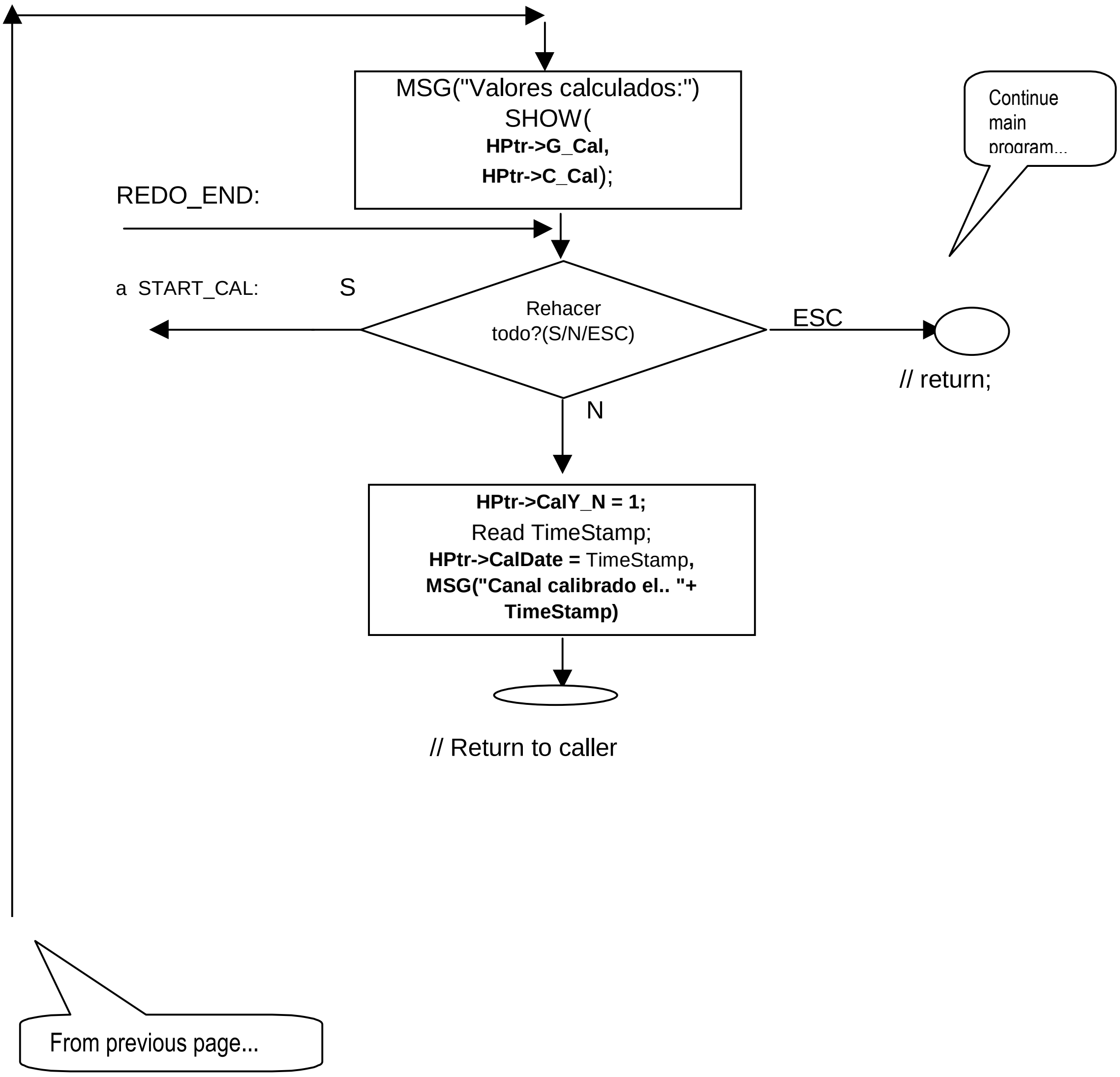
Teófilo de Loqui 58 (fdo)
9400 Rio Gallegos
Santa Cruz
ARGENTINA

L&R INGENIERIA
www.lyr-ing.com

TE: 54 (0) 2966 430923
FAX: 54 (0) 2966 430923 - 431081
e-mail: info@lyr-ing.com

ANNEX I
BLOCK DIAGRAM - CALIBRATION BY THE *TWO-POINT METHOD*





3. References

[Ref.1] *Embedded System Building Blocks, 2nd.Ed.*, Jean Labrosse, Ch. 10. R&D Books 2000, ISBN 0-87930-604-1