Freescale Semiconductor
Application Note

AN4193
Rev. 2.0, 11/2011

MM908E625 / MM908E626 - Factory Trimming and Calibration

1 Introduction

During final test of the MM908E625 / MM908E626, various parameters (such as, ICG trim value) are measured and stored into the device internal FLASH memory for internal and also for customer use. The purpose of these parameters is to enhance the accuracy of the MM908E625 / MM908E626 and reduce end of line calibration for the customer (for example, no ICG trimming necessary).

The following flash memory locations are reserved for this purpose and might have a value different from the "empty" (0xFF) state:

- 0xFD80:0xFDDF Trim and Calibration Values
- 0xFFFE:0xFFFF Reset Vector

In the event the application uses these parameters, one has to take care not to erase or override these values. If these parameters are not used, these flash locations can be erased and otherwise used.

Contents

2 General Description

2.1 Example Source Codes

For the C-source code examples the Metrowerks CodeWarrior was used. The assembly language examples are done in a generic way and are not specific to one Assembler. All source codes are just one way of implementing the trimming, and might or might not fit the requirements of a specific application.

The software is provided by Freescale “AS IS” and any expressed or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall Freescale or its contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) However caused and any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

2.2 FLASH Block Protection (page erase/mass erase)

The microcontroller core of the MM908E625 / MM908E626 is the MC68HC908EY16. The FLASH memory of this device has a built-in protection. This protection can protect the FLASH memory against erasing and overwriting. The amount of memory which is protected can be selected with the FLASH Block Protect Register (FBPR). By default, the last page (0xFFDC:0xFFFF) of the FLASH memory, containing the vector table, is always protected and can only be erased with Mass Erase. In the case that the trim values are used, the device needs special handling for FLASH programming, for example:

- •the FLASH area containing the trimming data has to be read
- •the device FLASH memory has to be erased (mass erase)
- •the FLASH has to be programmed with the program and the trimming data has to be reprogrammed

2.3 Available Trimming Data - Overview

The following table shows the user available trim data in the MM908E625 / MM908E626.

Table 1. Trim Data Memory Map (1)

	HOT	COLD	ROOM
ICG Trim 307.2kHz, 1%	n.a.	n.a.	0xFDC2
H-Bridge LS Current Recopy, HB1, Ratio High	0xFD83	0xFDA3	0xFDC3
H-Bridge LS Current Recopy, HB2, Ratio High	0xFD84	0xFDA4	0xFDC4
H-Bridge LS Current Recopy, HB3, Ratio High	0xFD85	0xFDA5	0xFDC5
H-Bridge LS Current Recopy, HB4, Ratio High	0xFD86	0xFDA6	0xFDC6
H-Bridge LS Current Recopy, HB4, Ratio Low	0xFD87	0xFDA7	0xFDC7
H-Bridge LS Current Recopy, HB4, Ratio Low	0xFD88	0xFDA8	0xFDC8
H-Bridge LS Current Recopy, HB4, Ratio Low	0xFD89	0xFDA9	0xFDC9
H-Bridge LS Current Recopy, HB4, Ratio Low	0xFD8A	0xFDAA	0xFDCA
VSUPRatio_TRIM	0xFD8B	0xFDAB	0xFDCB
H-Bridge LS Current Limitation Value CL1	0xFD90	n.a.	0xFDD0
H-Bridge LS Current Limitation Value CL2	0xFD91	n.a.	0xFDD1
H-Bridge LS Current Limitation Value CL3	0xFD92	n.a.	0xFDD2
H-Bridge LS Current Limitation Value CL4	0xFD93	n.a.	0xFDD3
H-Bridge LS Current Limitation Value CL5	0xFD94	n.a.	0xFDD4
Temperature Sensor Trim Value	0xFD95	0xFDB5	0xFDD5

Notes
1.Memory range for trim data 0xFD80..0xFDDF, all memory locations within that range not mentioned above should be ignored.

3 Functional Description

3.1 Internal Clock Generator (ICG) Trim Value

The internal clock generator (ICG) module is used to create a stable clock source for the microcontroller without using any external components.

With the MC68HC908EY16 data sheet, the trimming information for the ICG is part of the Microcontroller specification. Refer to the “ICG Trim Value” chapter for details.

3.2 H-Bridge LS Current Recopy

To determine the Current Recopy Trim value, the Current Sense Amplification is disabled, all HB Low Sides are switched on, and Current Recopy is sequentially selected as analog input for the ADC from HB1 to HB4. For each HB pin, 150 mA is injected into the HB pin, and after a wait time of 1.0 ms, a conversion is triggered. The same sequence is executed after enabling Current Sense Amplification (RATIO).

The Current Recopy Ratio measurements on HBx pins are converted to trim values using the following formula:

CRRatio = ((Digital Code x V_{REFH}) / 1023) / I_{LOAD}, where V_{REFH} = 5.0 V and I_{LOAD} = 150 mA

CRRatio_TRIM = (ref_value x 128) / CRRatio where ref_value = 2, when CSA = 0, and 12 when CSA = 1

The trim values (CRRatio_TRIM) for all three temperatures are stored in the FLASH memory location documented in Table 1.

3.3 VSUP Ratio

The following formula is applied to get the VSUPRatio value:

VSUPRatio = V_{SUP} / ((Digital Code x V_{REFH}) / 1023) where V_{REFH} = 5.0 V and V_{SUP} = 16 V

Based on this result, the trim adjustment is calculated:

VSUPRatio_TRIM = (ref_value x 128) / VSUPRatio where ref_value=5.1

The trim values (VSUPRatio_TRIM) for all three temperatures are stored in the FLASH memory location documented in Table 1.

3.4 H-Bridge LS Current Limitation Value

For each current limitation level, the values determined on all four low sides are averaged to determine the right trim adjustment through the following formulas:

- CL1_TRIM = (ref_value x 128) / AVERAGE_CL1_4HBs, where ref_value = 55 mA
- CL2_TRIM = (ref_value x 128) / AVERAGE_CL2_4HBs, where ref_value = 260 mA
- CL3_TRIM = (ref_value x 128) / AVERAGE_CL3_4HBs, where ref_value = 370 mA
- CL4_TRIM = (ref_value x 128) / AVERAGE_CL4_4HBs, where ref_value = 550 mA
- CL5_TRIM = (ref_value x 128) / AVERAGE_CL5_4HBs, where ref_value = 740 mA

The trim value for the hot and room temperature test is stored in the FLASH memory location documented in Table 1.

3.5 Internal Die Temperature Sensor - Offset Calibration Value

The internal die temperature sensor allows monitoring the analog die temperature. Due to the nature of the sensor, the linearity has little influence on process variations, but the offset of the sensor has a significant variance.

Table 2. Internal Die Temperature Slope

Voltage/Temperature Slope	S_{TTOV}	-	19	-	mV/°C
Output Voltage @ 25 °C	V_{T25}	1.7	2.1	2.5	V

As indicated in the datasheet, the output voltage at 25 °C can vary up to ±400 mV, which corresponds to an offset error of ±21 °C. The slope value is specified as typical only.

To compensate for this variation, the offset is measured during final test (3 temperatures) using the internal analog-to-digital converter, and stored into the FLASH memory. The conversion is performed as a 10-Bit AD conversion, the values are converted to 8-Bit accuracy before flashing. The values can be used by the application software to compensate the offset mentioned previously.

Table 3. shows the location and content of the FLASH temperature trimming information stored in the MC68HC908EY16.

Table 3. Temperature Offset Trimming Flash Content[1]

Flash Location	Register Content	Test Limits[2],[3]	Real Variation[2],[4]
$FD95	Hot Temperature Offset (N_{T135})	125 °C to 145 °C	133 °C to 138 °C
$FDB5	Cold Temperature Offset (N_{T-40})	-43 °C to -32 °C	-34 °C to -41 °C
$FDD5	Room Temperature Offset (N_{T25})	16 °C to 35 °C	26 °C to 31 °C

Notes
1.Stored as an 8-Bit ADC conversion result
2.Device Temperature during final test trimming, VSUP = 16 V
3.Maximum variation allowed during production
4.Three production lots with different testers

Offset Compensation Usage using the Room Temperature Offset (N_{T25}):

The analog die temperature *ChipTemp* can be calculated with the following equation

$$\mathrm{ChipTemp} = 25°\mathrm{C} + \frac{V - V_{T25}}{S_{TtoV}}$$

with the measured voltage of the analog-to-digital voltage *V*, the reference offset voltage at 25 °C V_{T25} and the temperature to the voltage slope (sensor voltage resolution) S_{TtoV} of the sensor. For the chip temperature offset V_{T25}, the stored calibration value can be used instead using the typical value from the datasheet.

The measured voltage V_{ADC} can be evaluated from the actual measured analog-to-digital converter value N_{ADC} and the analog-to-digital resolution R_{ADC} with:

$$V_{ADC} = N_{ADC} \cdot R_{ADC}$$

The analog-to-digital resolution R_{ADC} for an 8-bit (truncated) conversion is:

$$R_{ADC} = \frac{5V}{255} \approx 19,6\mathrm{mV}$$

Substituting the voltage *V* by the analog-to-digital value *N* leads to the following expression.

$$\mathrm{ChipTemp} = 25°\mathrm{C} + \frac{R_{ADC}}{S_{TtoV}} \cdot (N - N_{T25})$$

which is the basis for the calculations of the chip temperature by the embedded software. With analog to digital converter “temperature resolution”:

$$\frac{R_{ADC}}{S_{TtoV}} = \frac{\frac{5V}{255}}{19\frac{mV}{°C}} \approx 1,032°\mathrm{C}$$

Therefore the chip temperature *chip_temp* can be calculated as:

$$\text{ChipTemp} = 25°C + 1,032°C \cdot (N - N_{T25})$$

***Method 1** - exact calculation using float datatyp*

C language example

```
#include<stdlib.h>

// --------------------------------------------------------
// -- DEFINES --
// --------------------------------------------------------
#define IOBYTE(address) (*((volatile unsigned char*) (address)))
#define IOWORD(address) (*((volatile unsigned int*) (address)))
#define N_T25I          OBYTE(0xFDD5)                     // temperature offset calibration value (25C)
#define S_TtoV          0.019                             // 19mV per celsius
#define R_ADC           (5.0/255)                         // ADC resolution (8bit)
// --------------------------------------------------------
// -- MAIN --
// --------------------------------------------------------
void main(void)  {

    unsigned char  N;                                     // measure adc value (8-bit truncated mode)
    float ChipTemp;


    ......

    N = ADC_Value(xxx);                                   // measure temp sensor voltage with adc
    ChipTemp =  25 + (R_ADC/S_TtoV) * (N - N_T25);        // calculate chip temp

    ......

}
```

***Method 2** - approximation using integer datatyp*

With an approximation for R_{ADC} / S_{TtoV25} = 1 the chip temperature ChipTemp simplifies to

$$\text{ChipTemp} = 25°\text{C} + (N - N_{T25})$$

The relative error through the approximation is ~ 3.2% or 0.032 °C/bit. With the reference at 25 °C, the maximum absolute error for the range of -40 °C to +125 °C is

- 2.08 °C at -40 °C and
- -3.2 °C at +125 °C

As this is a know static error (systematic error), it also could be compensated by the software.

C language example

```
#include<stdlib.h>

// -------------------------------------------------------
// -- DEFINES --
// -------------------------------------------------------
#define IOBYTE(address) (*((volatile unsigned char*) (address)))
#define IOWORD(address) (*((volatile unsigned int*) (address)))

#define N_T25        IOBYTE(0xFDD5)                  // temperature offset calibration value (25 °C)

// -------------------------------------------------------
// -- MAIN --
// -------------------------------------------------------
void main(void)  {

    unsigned char  N;                                // measure adc value (8-bit truncated mode)
    int ChipTemp;


    ......

    N = ADC_Value(xxx);                              // measure temp sensor voltage with adc
    ChipTemp =  25 + (N - N_T25);                    // calculate chip temp

    ......

}
```

Comparison of the two methods

Table 4. give a comparison between the two methods. It indicates the ROM, RAM/Stack size necessary just for the calculation (only the code line with the formula is considered), and compares the execution time for the calculation operation.

Table 4. Performance Data

	Method 1 - “Float”	Method 2 - “Integer”
ROM size	1161	30
RAM (ChipTemp) / Stack usage	4 / 44	2 / 3
Cycles for calculation	4289	13
Execution time @5.0 MHz bus frequency	857.8 µs	2.6 µs

The performance requirements for the Method 1 are significantly higher, for example, the execution time of method 1 is more than 300 times longer, the ROM memory is 38 times higher, and the RAM memory is 9 times higher.

NOTE

The example given is using only one temperature point for calibration. Additional accuracy can be reached, including all three available points to calculate the device specific slope. It is recommended to calculate two slope factors for (-40…25 °C and 25 °…135 °C) based on the given trim values.

4 Comment / Summary

Based on the calibration method and the trim values used, a significant increase of accuracy of the MM908E625 / MM908E626 can be realized.

5 References

1. MC68HC908EY16
2. MM908E625 Data Sheet
3. MM908E626 Data Sheet

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

AN4193
Rev. 2.0
11/2011