Freescale Semiconductor
Application Note

Document Number:
Rev. 1.0, 4/2013

CPU, Flash and RAM Benchmarks

Xtrinsic eCompass and Magnetic Calibration Algorithms

by: Mark Pedley

Contents

1 Introduction

This Application Note contains measured benchmarks for Freescale's Xtrinsic eCompass and magnetic calibration software provided under license at www.freescale.com/ecompass. This Application Note is part of the technical documentation for that software and its use and distribution are controlled by that license agreement.

1.1 Summary

This document contains the program memory (flash), working memory (RAM) and floating point requirements of the Freescale Xtrinsic eCompass and calibration algorithms.

Figures are presented for all three coordinate systems (Aerospace NED, Android and Windows 8) and for all magnetic calibration solvers (10-, 7-, and 4-element). The final product will, however, only require one coordinate system and one calibration solver.

Program memory (flash) figures were obtained by compiling the code for the ARM Thumb2 instruction set. The original ARM instruction word length was 32 bits. The ARM Thumb instruction set is a limited 16-bit instruction set designed to reduce the program memory footprint. The current ARM Thumb2 instruction set is intended to provide the best of both worlds by using both 16-bit and 32-bit instruction word lengths. Users of conventional 32-bit processors should probably double the program memory size figures in this document.

The flash memory requirement for ARM Thumb2 processors should be under 10 kB not including floating point emulation libraries. This comprises approximately 1.5 kB in overall control software, 3.5 kB to 4 kB for eCompass and orientation functions and 4 kB to 5.3 kB for the magnetic calibration functions.

The RAM requirement should be under 4 kB irrespective of the magnetic calibration function algorithm used since storage is dominated by the magnetometer measurement buffer which is common to all magnetic calibration solvers.

The software makes intense use of floating point operations in the magnetic calibration functions. This should cause no performance issues on processors with a built-in hardware floating point unit. The software has, however, been extensively optimized to minimize the number of floating point operations to allow use on integer processors using software floating point emulation libraries. On integer processors, 90% or more of the processing will be on floating point calculations and less than 10% on integer operations.

The flash memory footprint of floating point emulation libraries is processor dependent and not included in the figures in this document.

Users should budget for approximately 300 floating point operations per iteration of the eCompass sampling loop which typically occurs at a rate of 20 Hz to 50 Hz. The magnetic calibration software requires approximately 3000, 20,000 and 60,000 floating point operations per execution for the 4-, 7-, and 10-element magnetic calibration code respectively but since this function need only execute every few thousand iterations of the eCompass, the average floating point computation load is significantly less than for the eCompass.

2 Program Memory (Compiled for ARM Thumb2)

2.1 Main control loop

ARM Thumb2 bytes	All solvers and all coordinate systems
main()	1530
Total	1530

2.2 Orientation functions

ARM Thumb2 bytes	NED	Android	Win8
feCompassDirectNED	854		
fNEDAnglesDegFromRotationMatrix	290		
feCompassDirectAndroid		862	
fAndroidAnglesDegFromRotationMatrix		296	
feCompassDirectWin8			872
fWin8AnglesDegFromRotationMatrix			708
fQuaternionFromRotationMatrix	630	630	630
fInitLPFOrientationMatrix	226	226	226
fLPFOrientationMatrix	732	732	732

Table continues on the next page...

ARM Thumb2 bytes	NED	Android	Win8
fmatrixAeqRenormRotA	508	508	508
Total	**3240**	**3254**	**3676**

2.3 Calibration functions

ARM Thumb2 bytes	10-element	7-element	4-element
fUpdateCalibration10EIG	1804		
fUpdateCalibration7EIG		1508	
fUpdateCalibration4INV			1622
eigencompute	1080	1080	
f3x3matrixAeqInvSymB	466		
fmatrixAeqRootSymB	118		
fmatrixAeqInvA			608
fInvertMagCal	262	262	262
fUpdateMagnetometerBuffer	662	662	662
ResetMagCalibration	114	114	114
fmatrixAeqBxC	114	114	114
fmatrixAeqBxTrC	114	114	114
fmatrixAeqBxTrB	126	126	126
fmatrixAeqI	54	54	54
fmatrixAeqAxScalar	56	56	56
fmatrixAeqMinusA	60	60	60
f3x3matrixDetA	168	168	168
fmatrixAeqB	54	54	54
fmatrixAeqScalar	46	46	46
Total	**5298**	**4418**	**4060**

3 Working Memory (RAM)

3.1 Orientation functions

Structure and RAM (bytes)	NED	Android	Win8
thisAccel	24	24	24
thisMag	60	60	60

Table continues on the next page...

Structure and RAM (bytes)	NED	Android	Win8
thisSV6DOF	376	376	376
Total	**460**	**460**	**460**

3.2 Calibration functions

RAM usage is dominated by the magnetometer buffer providing measurements for the calibration algorithms and to a lesser extent by the working matrices used by the magnetic calibration algorithms. If the RAM footprint must be reduced, the magnetic buffer could be reduced in size to 5x5x5 elements with a corresponding change in the compile time constants controlling the magnetic calibration.

```
// default settings for 6x6x6 magnetic buffer
#define MINEQUATIONS 24
#define MEDEQUATIONS 80
#define MAXEQUATIONS 120

// recommended settings for 5x5x5 magnetic buffer
#define MINEQUATIONS 24
#define MEDEQUATIONS 64
#define MAXEQUATIONS 96
```

For reasons discussed elsewhere in the documentation, matrices are accessed via a one dimensional array of pointers to rows of the underlying matrix. This explains the appearance of two figures in the table for each matrix with the sum of these two figures used in the columns for each algorithm.

Structure and RAM (bytes)	Size	Pointers	Size	10-element	7-element	4-element
thisMagCal	236			236	236	236
thisMagneticBuffer (size 6x6x6)	2164			2164	2164	2164
xftmpA10x10	400	ftmpA10x10	40	440		
xftmpB10x10	400	ftmpB10x10	40	440		
xftmpA10x1	40	ftmpA10x1	40	80		
xftmpA7x7	196	ftmpA7x7	28		224	
xftmpB7x7	196	ftmpB7x7	28		224	
xftmpA7x1	28	ftmpA7x1	28		56	
xftmpA6x1	24	ftmpA6x1	24			
xftmpA4x4	64	ftmpA4x4	16			80
xftmpB4x4	64	ftmpB4x4	16			80
xftmpA3x3	36	ftmpA3x3	12	48		
xftmpA4x1	16	ftmpA4x1	16			32
xftmpB4x1	16	ftmpB4x1	16			32
xftmpA3x1	12	ftmpA3x1	12	24		
xicolind	24	icolind	24			48
xirowind	24	irowind	24			48
xipivot	24	ipivot	24			48
Total				**3432**	**2904**	**2768**

4 Floating Point Operations

4.1 Orientation functions

This table summarizes floating point operations for each iteration of the eCompass which will typically be between 20 Hz and 50 Hz. This includes the two functions fInvertMagCal to remove the hard and soft iron distortion from the current measurement and fUpdateMagnetometerBuffer to place the current magnetometer measurement in the correct position in the magnetometer measurement buffer.

Floating Point Operations	NED	Android	Win8
fInvertMagCal	~18	~18	~18
fUpdateMagnetometerBuffer	~12	~12	~12
feCompassDirectNED	~75		
fNEDAnglesDegFromRotationMatrix	~6		
feCompassDirectAndroid		~75	
fAndroidAnglesDegFromRotationMatrix		~6	
feCompassDirectWin8			~75
fWin8AnglesDegFromRotationMatrix			~15
fQuaternionFromRotationMatrix	~20	~20	~20
fLPFOrientationMatrix	~100	~100	~100
fmatrixAeqRenormRotA	~36	~36	~36
Total	**~267**	**~267**	**~276**

4.2 Calibration functions

The 10- and 7-point calibration algorithm processing is dominated by the 10x10 and 7x7 matrix eigenvalue and eigenvector calculation in function eigencompute. The measured benchmarks in the table are for 96 measurements in the magnetic buffer but will be very similar for more or fewer measurements.

Floating Point Operations	**Total every execution**
fUpdateCalibration10EIG (96 measurements)	61509 (49322 of which is eigencompute)
fUpdateCalibration7EIG (96 measurements)	19891 (13599 of which is eigencompute)
fUpdateCalibration4INV (96 measurements)	3296

The calibration algorithms require a significant number of floating point calculations but do not dominate the processing since the calibration need only run intermittently. The calibration mostly changes as a result of temperature changes in the smartphone resulting from its own temperature dissipation or from changes in the ambient temperature. Assuming a thermal time constant of one minute and an eCompass sampling rate of 50 Hz then the calibration might need to only run every 3000 iterations. The average floating point overhead per iteration is then only approximately 20 operations (61509 divided by 3000).

For simplicity, the reference software is single threaded and calls the calibration algorithm every `INITIALCALINTERVAL` or `FINALCALINTERVAL` iterations. This suffices to demonstrate the principle of the software but has the drawback of introducing a regular timing interruption although, in practice, this is be noticeable. Users may wish to consider running the calibration as a low priority thread in parallel with the eCompass software. This is discussed in detail in Application Note AN4700.

4.3 Floating point emulation libraries

If your processor does not have a hardware floating point unit (FPU) then your build will link in software emulation libraries. The size of this code is processor dependent but 35 kB has been measured by Freescale on builds for ARM Thumb2 processors. These libraries will only increase the size of your build if your product's software does not already perform floating point calculations.

How to Reach Us:

Home Page:
www.freescale.com

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

USA/Europe or Locations Not Listed:
Freescale Semiconductor
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

Document Number:
Rev. 1.0, 4/2013

freescale™