Document Number: ISF2P1_KINETIS_RELEASE_RN
Rev. 0, 3/2015

Intelligent Sensing Framework V2.1 for Kinetis, Release Notes

Contents

freescale Sensors

1 Introduction

This document describes the Freescale Intelligent Sensing Framework version 2.1 (ISF v.2.1) middleware, released for Kinetis platforms. ISF v2.1 consists of Processor Expert components that take advantage of Processor Expert's capability to configure any of the Kinetis processors supported by Logical Device Drivers in Processor Expert.

2 Requirements

The Intelligent Sensing Framework (ISF) version 2.1 middleware for Kinetis has several requirements for tools, development systems and deployment targets.

2.1 Development Tools

The ISF v2.11 Release was compiled and tested with the following development tools with built-in support for Kinetis using Freescale's Processor Expert technology:

- The latest update for Windows of the CodeWarrior Development Studio Version 10.6 for microcontrollers
- Kinetis Development Studio (KDS) Version 2.0

2.2 System Requirements

The ISF implementation for the Kinetis platforms accommodates a wide variety of host system configurations.

Table 1. System Requirements for ISF v2.1

Parameter	Minimum PC Configuration	Recommended PC Configuration
Operating system	Windows 7	
Communications to target hardware	USB port	
Processor speed (GHz)	1.8	2.6
RAM (GB)	2	4
Free disk space (GB)	20	400

2.3 Generic Support of Kinetis and Sensors

The ISF v2.1 executes on a Kinetis MCU and supports a variety of sensor types. The Kinetis MCUs supported in this release include the Kinetis implementations of the ARM® Cortex® M0+, the ARM® Cortex® M4, and the ARM® Cortex® M4F. The generic sensors types supported in ISF v2.1 are accelerometers, magnetometers, gyroscopes, pressure sensors, orientation sensors, and temperature sensors.

2.3.1 Tested Implementations

The ISF v2.1 release supports the Freescale Freedom Development platforms for Kinetis KL02 MCUs, FRDM-KL25Z, FRDM-K22F, and the FRDM-K64F, which are typical boards for the KL25 (ARM® Cortex® M0+ Core), the K22F(ARM® Cortex® M4F Core), and the K64F (ARM® Cortex® M4F Core), respectively. In addition, there are a number of existing and future sensor shield boards that can be used in conjunction with the Freescale Freedom boards including:

- FRDM-FXS-MULTI, a Freescale Freedom Development board containing a variety of Freescale sensors and connectors for assembling the two boards.
- FRDM-FXS-MULTI-B, a Freescale Freedom Development board with Bluetooth, containing a variety of Freescale sensors and connectors for assembling the two boards.
- FRDM-STBC-AGM01, a Freescale Freedom Development board containing the Freescale 3-axis gyroscope (FXAS21002C), and the Freescale 6-axis accelerometer/magnetometer (FXOS8700CQ). Various kits are available providing both the Freescale Freedom board and the shield board.
- FRDM-STBC-xxxx, future shield boards containing various Freescale sensors supporting a variety of applications.
- The Freescale Sensors supported in the release according to sensor type are listed in Table 2

Table 2. Freescale Sensors supported in this release

Sensor Type	Orderable Part Number (with Links to Freescale.com)
Accelerometer:	MMA8652, MMA8653, FXOS8700C, FXLS8471Q
Magnetometer:	MAG3110, FXOS8700C
Gyrometer	FXAS21002C
Pressure	MPL3115A2
Orientation	Sensor Fusion (any combination of accelerometer, gyroscope, or magnetometer)

3 Release Content

Table 3 provides a summary of the contents of the ISF2.1 installer. These items are also included on the Downloads tab of the ISF website located at www.freescale.com/isf

Table 3. ISF Downloads

Run-time Software	Status
Middleware-Framework	
ISF2P1_Installer Installer for ISF v2.1 documentation, tools, and project files There is an “uninstall” feature included with the installer located in the Start menu with the other ISF 2.1 artifacts.	Updated
ISF2P1_PEx.PEupd	Updated
Application Specific-Apps, also known as, Example Applications	
ISF_KL25Z_CW_PROJ Basic Code Warrior Project using ISF 2.1 with a KL25Z MCU. Register Level Interface capability is included.	Updated

Release Content

Run-time Software	Status
ISF_KL25Z_KDS_PROJ Basic KDS Project using ISF 2.1 with a KL25Z MCU. Register Level Interface capability is included.	New
ISF_K22F_KDS_PROJ Basic KDS Project using ISF 2.1 with a K22F MCU. Register Level Interface capability in included.	New
ISF_K22F_KDS_SF_PROJ KDS Project using ISF 2.1 on a K22F MCU running sensor fusion.	New
ISF_K64F_CW_PROJ Basic Code Warrior Project using ISF 2.1 with a K64F MCU. Register Level Interface capability is included.	Updated
ISF_K64F_KDS_PROJ Basic KDS Project using ISF 2.1 with a K64F MCU. Register Level Interface capability is included.	New
ISF_K64F_CW_SF_PROJ CW Project using ISF 2.1 on a K64F MCU running sensor fusion.	New
ISF_K64F_KDS_SF_PROJ KDS Project using ISF 2.1 on a K64F MCU running sensor fusion.	New

The ISF Processor Expert component is contained within the ISF2P1_PEx.PEupd file loaded from the ISF web site into Processor Expert via CodeWarrior or KDS. The component library contains the following components:

Table 4. ISF Component Status

Implicit Processor Expert Components contained within ISF_Core Component	Status
ISF_CommChannel_I2C	Updated
ISF_CommChannel_SPI	Updated
ISF_CommChannel_UART	Updated
ISF_CommChannelConfig	Updated
ISF_Core	New
ISF_Protocol_Adapter	Updated
ISF_RLI (Register Level Interface)	Updated
ISF_Sensor_FXAS21002_Gyrometer	Updated
ISF_Sensor_FXLS8471_Accelerometer	New
ISF_Sensor_FXOS8700_ECompass	Updated
ISF_Sensor_MAG3110_Magnetometer	Updated
ISF_Sensor_MMA865x_Accelerometer	Updated
ISF_Sensor_MPL3115_Pressure	New
ISF_Sensor_Virtual_Orientation	New
ISFBusManager	Updated
ISFEmbApp	New

Release Content

In addition, the ISF v2.1 installer (ISF2.1_Installer) includes the following documentation, which is also available individually from the Documentation tab of the Freescale ISF web site.

Table 5. ISF documentation status for v. 2.1

Documentation	Status
ISF2P0_KINETIS_SW_REFERENCE_RM Software Reference Manual for the Intelligent Sensing Framework v2.0 Reference Manual	Existing
ISF2P1_KINETIS_API_REFERENCE_RM API Reference Manual for the Intelligent Sensing Framework (Unzip and open ISF_2.1_API_Reference.html)	Updated
ISF2P1_KINETIS_RELEASE_RN Software Release Notes for the Intelligent Sensing Framework v2.1	Updated

Furthermore, the ISF v2.1 installer includes training videos, which are also available individually from the Training tab of the Freescale ISF web site. Check the ISF website for additional videos provided post release.

Table 6. ISF Training resources status

Training	Status
ISF R2.0_Hardware and Software Setup Training Video showing how to get started with ISF 2.0 using CodeWarrior	Existing
ISF R2.0 Installation	Existing
ISF R2.0 FRDM-K64F Example Project (uses CodeWarrior)	Existing
KIT V2 Tool Usage – Data Analysis (KIT V2 Tool is included in Freedom Sensor Toolbox)	Existing
KIT V2 Customizing the Graphical Display (KIT V2 Tool is included in Freedom Sensor Toolbox)	Existing
ISF R2.0 – Adding Your Code to the Sample Applications Provided	Existing
ISF R2.0 – Sensor Component Register Settings (How to configure the sensor components)	Existing
ISF R2.0 – Creating Host Interface Commands	Existing
ISF R2.0 – Getting started with your own data formats using the Streaming Protocol	Existing
ISF R2.1 – Product Overview and Demo	New
ISF R2.1 – Orientation Sensor Demo	New
ISF R2.1 – Using the Register Level Interface	New

4 What is New for ISF v2.1 Release

In addition to some ease of use items, the core libraries have been replaced with “public” source code.

The KIT V2 is now included as one of two options that may be invoked from the Freedom Sensor Toolbox Launcher. The Freedom Sensor Toolbox Launcher incorporates both the KIT V2 Tool and the Freescale Sensor Fusion Toolbox. Once invoked, the KIT V2 Tool operates exactly the same as for ISF v2.0. This allows all the materials from the ISF v2.0 videos to apply directly to ISF v2.1. The Freedom Sensor Toolbox Launcher can also optionally invoke the Freescale Sensor Fusion Toolbox which displays sensor fusion output for the sensor fusion projects associated with the FRDM-K22F and FRDM-K64F boards.

Sensor Fusion is available through the virtual orientation sensor which combines data from available and configured accelerometer, magnetometer, and/or gyroscope. This is an ISF implementation of the Freescale Sensor Fusion library.

Additional sensors supported include the MMA865x (MMA8652 and MMA8653), FXLS8471Q, and MPL3115A2. The updated list of sensor adapters are:

Sensor Adapters using generic APIs based upon sensor types:

- Accelerometer
  - — FXLS8471
  - — FXOS8700C
  - — MMA865x (MMA8652 and MMA8653)
- Gyroscope
  - — FXAS21002C
- Magnetometer
  - — FXOS8700C
  - — MAG3110
- Pressure
  - — MPL3115
- Orientation (Virtual)
  - — Sensor Fusion

4.1 ISF v2.0 Features

The middleware is integrated into the Freescale Processor Expert technology. This allows the middleware to be autogenerated for specific hardware configurations.

- By using Processor Expert, ISF is abstracted from the specific MCU hardware. ISF can be configured for almost all of the Kinetis processors, those with a PIT timer. Implementations for two Kinetis processors, the KL25Z and the K64F, have been tested and verified for this release.
- The middleware includes a set of Processor Expert components downloadable from the Freescale ISF web site.

The ISF_Core Component is a library of components. Some of the components of interest within the ISF_Core library are:

Protocol Adapters

- I2C for sensor communication
- SPI for sensor communication
- UART for host communications

Sensor Adapters using generic APIs based upon sensor types

- Accelerometer
  — FXOS8700C
  — MMA865x
- Gyroscope
  — FXAS21002C
- Magnetometer
  — MAG3110
  — FXOS8700C

Streaming protocol allows wider options in communicating sensor data back to the host. CRC error checking is included as an optional feature.

DSA-Direct API replaces the functionality of the Sensor Manager and can manage sensor data directly. Register level access to the sensors is provided. Optimization of memory usage is implemented.

4.2 ISF v1.1 Features

The following features from ISF v1.1 are available in the new release.

- The middleware acts as a sensor hub, providing sensor data from external sensors.
- CodeWarrior project files are available allowing users to begin with a working project.
- Automated installation is provided to enable ease of use.
- Synchronization is now provided during framework initialization to ensure that initialization is complete prior to executing application code.
- Application-specific commands are supported by the Command Interpreter.

5 Installation Instructions

A single, unified installer is provided for installing the Documentation, Training and Example Projects. ISF artifacts can be installed by going to www.freescale.com/isf and clicking on the “Download” box.

The Processor Expert PEupd file is a separate download as it will be updated with new sensor adapters between releases. It can be downloaded from the homepage or the Featured Downloads tab. Once downloaded, install the Processor Expert PEupd file within the Kinetis Design Studio (SKD) or Code Warrior using the Processor Expert pull down menu, select Import Component, choose the peupd package and click Open.

Refer to the Freescale Processor Expert web site to learn about Processor Expert and how to add components, including the ISF components to a project. The example projects provided may be used as examples to provide users a starting point.

6 Release Overview

This is the Intelligent Sensor Framework (ISF) release by Freescale Semiconductor. It targets the Freescale Kinetis platforms using a PIT timer. ISF is built upon the Freescale MQX™ Lite RTOS, and supported by CodeWarrior.

6.1 Using and modifying the ISF source, project or template files

The ISF release makes source, project, and template files available. All of the ISF2P1 CodeWarrior and KDS projects are provided as user applications, sample code for users to adapt and modify.

NOTE: Freescale is not responsible for the support of any modified examples or templates.

7 Memory Footprint for Target: Kinetis K64F

Table 7. ISF Memory requirements

Component	Code Size (bytes)				Description
	Flash		SRAM		
	DEC	HEX	DEC	HEX	
Required ISF Components					
MQX™ Lite RTOS	14626	3922	48	30	Minimum MQX Kernel
Processor Expert LLD Components	6848	1AC0	594	252	Logical Device Drivers used by Processor Expert
ISF_Core	1232	4D0	6	6	ISF Initialization and Configuration functions
Optional ISF Components					
ISFBusManager	1470	5BE	116	74	Component
			1024	400	Task Stack Space
ISF_RLI	644	284	532	214	Register Level Interface
ISFEmbApp	846	34E	332	14C	Component
			1280	500	Task Stack Space
Protocol Adapter Components					
ISF_Protocol_Adapter	0	0	0	0	Abstract Component
ISF_CommChannelConfig	0	0	0	0	Abstract Component
ISF_CommChannel_I2C	1332	534	0	0	Component
ISF_CommChannel_SPI	1102	44E	0	0	Component
ISF_CommChannel_UART	1126	466	0	0	Component
Sensor Adapter Components					
ISF_Sensor_FXAS21002_Gyrometer	1830	726	20	14	Gyroscope
ISF_Sensor_FXLS8471_Accelerometer	2046	7FE	173	AD	Accelerometer using SPI Protocol
ISF_Sensor_FXOS8700_ECompass	1991	7C7	20	14	Accelerometer and Magnetometer

Component	Code Size (bytes)				Description
ISF_Sensor_MAG3110_Magnetometer	1542	606	16	10	Magnetometer
ISF_Sensor_MMA865x_Accelerometer	1634	662	17	11	Accelerometer
ISF_Sensor_MPL3115_Pressure	1860	744	72	48	Pressure Sensor
ISF_Sensor_Virtual_Orientation	2706	A92	388	184	Sensor Fusion
*Sensor Fusion*1	19912	4DC8	8152	1FD8	Dynamic memory
			2000	7D0	MagCal (Magnetic Calibration) Stack Space
			2000	7D0	Fusion Stack Space
ISF Sensor Fusion Embedded Application	56528	DCD0	26682	683A	Size of large Sensor Fusion application
Minimum and Maximum ISF Configuration Sizes					
Minimum ISF Configuration including MQX™ RTOS	34427	867B	5590	15D6	This is the minimal required configuration of ISF.
Maximum Space Available for Customer	1014149	F7985	256554	3EA2A	This is the maximum memory available to customers for application development.

1. Sensor Fusion demo application that communicates with the Sensor Fusion Toolbox PC GUI configured to use FXOS8700 and FXAS21002 sensor adapters running six fusion algorithms (Tiltmeter, 2D Automotive Compass, Rotation, Tilt-Compensated Compass, Gaming Handset, and Gyro-Stabilized Compass)

NOTE: Optimization is set to –Os to optimize for code size. While the table above provides the minimum available memory sizes, the available memory is generally based upon the configuration of ISF, the number of sensor adapters, and the complexity of the algorithms involved.

8 CPU Load

The computational load imposed on a system by different applications that use ISF components, may be different and it is not possible to measure every instance or configuration. To estimate the CPU Load demanded by an application that uses ISF on the K64F Freescale Freedom Board, the time required for ISF to process one sensor data sample is used The measurement of computational load begins when the ISF Bus Manager (BM) PIT timer generates an interrupt. The measurement stops when the user- embedded application receives notification that sensor data is available. Key aspects of the measurement include:

CodeWarrior tool v10.6.1 and the following ISF components were used:

- DSA Direct, Bus Manager (BM), and the FXAS21002 sensor adapter
- MQX™ RTOS Lite embedded in CodeWarrior v10.6.1 as a Processor Expert Component
  — The source code was compiled with optimization set for size in order to provide the smallest code size

A key peripheral of the K64F hardware that is actively running is the PIT timer, used by the Bus Manager. Each timer count is 1 µsec, as configured by the Bus Manager during initialization. ISF must be configured to run at 120 MHz core and 24 MHz bus in order for the PIT timer to run at 1 µsec per timer count. The PIT timer is considered to be the most accurate method for measurement because the Bus Manager establishes the PIT timer, creating interrupts at the sample rate requested by the user. It restarts at a count of zero, after an interrupt is generated. After the sensor data is retrieved by the user's task, the PIT counter value reflects the time required for ISF to perform the following activities:

The BM includes processing the PIT interrupt

- BM Task switch and invocation of the sensor adapter callback
- The sensor adapter callback retrieves the sensor data and includes serial communication with the sensor and sets an event flag, that notifies the application
- Task switches to the user embedded application, after a notification of the availability of sensor data

The test application is generated using the standard ISFEmbApp PEx component with a FXAS21002 subscription. The initialization section of the embedded application is modified to configure the application to subscribe to sensor data and start running. A sample counter was added to the application loop, and increments with the receipt of each new sensor sample. Finally, a time stamp variable stores the PIT counter value with the receipt of each new sensor sample.

In addition, a test task was created to run at the lowest priority of all other tasks. It runs as a loop that blocks on the Wait for Interrupt (WFI) instruction. This test task unblocks when an interrupt for the TPM timer or the MQX systick timer occurs. It then checks the sample counter to determine if a new sample has been received. If a new sample has been received, the time stamp value is added to an accumulator variable. Otherwise, the WFI instruction is executed. The loop runs for a specified number of times, before exiting. The test setup ran for 2000 samples.

Table 8. CPU Load Test Parameters

Speed (MHz)	Sample Count	Total time in (µsec)	Average (µsec/sample)	Average latency for I^2C (µsec/sample)	Average latency for ISF (µsec/sample)
120 MHz ARM® Cortex®core frequency	2000	628000	314	290	24
24 MHz bus clock frequency					
24 MHz flash clock frequency					

The average latency time is the accumulated time divided by the sample counter value.

A limitation of the method used is that the sample rate cannot be set faster than the time it takes for ISF and the user application to retrieve sensor data.

The measured time for both the ISF components and the user task to retrieve sensor data and be ready for algorithm processing, averaged 314 µsec. Out of the 314 µsec latency, 290 µsec were due to serial communication with the sensor. Therefore, ISF code latency is 24 µsec.

Table 9. Measured CPU Load

Frequency (Hz)	Period (Msec)	ISF CPU Load (%)	CPU Load Total (%)
12.5	80	0.04	0.4
25	40	0.06	0.78
50	20	0.12	1.57
100	10	0.24	3.14
200	5	0.48	6.28
400	2.5	0.96	12.56
800	1.25	1.92	25.12

9 Known Issues and Limitations

9.1 Compiler/IDE issues

By default, the CodeWarrior tool suite uses unsigned, 8-bit integers for its char data type. The user may consider this to be unexpected behavior. It may be changed by unchecking the Use Unsigned Chars compiler configuration option in the Project > Properties > C/C++ Build > Settings > ARM Ltd Windows GCC C Compiler > Miscellaneous Settings page.

9.2 Known software issues

Each embedded application is limited to accessing each specific sensor in a single subscription.

9.2.1 Open Defects

Table 10. Open defects

Defect ID	Ticket Summary	Ticket Submission Date	Priority
CR340207	The installation shows unit test information released in the core library.	18 November 2014	L3
CR340211	ISF R2.0 depends upon an update to the MQXLite_task PEx component in order to compile properly. Users must use the most recent version of CW10.6.1.	18 November 2014	L2
CR340214	ISF Sensor Adapter PEx Components sometimes fail to offer I2C_CH1 Communications Channel to User.	18 November 2014	L3
CR340662	In order for the CI Streaming feature to work properly, the user must manually type in the following into the Protocol	21 November 2014	L3

Defect ID	Ticket Summary	Ticket Submission Date	Priority
	component methods: ci_stream_init ci_protocol_CB_stream		
SSDSBOX-30	Some internal tasks using floating point calculations do not enable MQX_FLOATING_POINT_TASK properly.	21 January 2015	L3
SSDSBOX-49	There are numerous typographical errors in the generic sensor type definitions	22 January 2015	L5

9.2.2 Closed Defects

Table 11. Closed defects

Defect ID	Ticket Summary	Ticket Closure Date	Priority
CR345614	The CI mailbox protocol does not use 2 bytes for offset as specified in the SWRM	30 January 2015	L3
CR345907	If the Component name for the ISFEmbApp component is changed, the compilation fails.	2 February 2015	L3
CR345910	The FXOS8700CQ sensor adapter does not properly support accelerometer only usage at frequencies other than 400 Hz.	30 January 2015	L3
CR346036	The Device ID Command does not consistently return the expected 18 bytes of data.	30 January 2015	L2
SSDSBOX-81	Timestamp resolution has only 5 msec resolution by default and may create duplicate timestamps.	9 February 2015	L3

10 ISF Software Change Log

Table 12. ISF Version Changes

Version 2.1 (March 2015)
Sensor Fusion incorporated as a virtual Orientation sensor.
Version 2.0 (December 2014)
ISF integrated with Processor Expert with hardware abstraction for entire Kinetis platforms supporting PIT timers
Version 1.1 (April 2014)
This is the initial ISF release supporting Kinetis KL25Z.

11 Revision History

Rev. No.	Date	Description
0	3/2015	Initial Public Release

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Document Number: ISF2P1_KINETIS_RELEASE_RN
Revision 0, 3/2015