Document Number: ISF2P2_KINETIS_RELEASE_RN
Rev. 2.0, 10/2016

Intelligent Sensing Framework Version 2.2 for Kinetis Release Notes

Contents

1 Introduction

This document describes the NXP Intelligent Sensing Framework version 2.2 (ISF v2.2) middleware, released for Kinetis platforms. ISF v2.2 consists of Processor Expert components that take advantage of Processor Expert's capability to configure the Kinetis processors supported by the Kinetis Software Development Kit (KSDK) in Processor Expert.

2 Requirements

ISF v2.2 middleware for Kinetis has several requirements for tools, development systems and deployment targets.

2.1 Development tools

The ISF v2.2 release was compiled and tested with the following development tools with built-in support for Kinetis using NXP's Processor Expert technology:

- Kinetis Development Studio (KDS) Version 3.2
- Kinetis Software Development Kit (KSDK) Version 1.2

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 v2.2

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 v2.2 executes on a Kinetis MCU and supports all sensor types with the exception of those requiring 5 V analog output and the generic data logger application.

2.3.1 Tested implementations

The ISF v2.2 release supports the NXP Freedom boards: FRDM-KL25Z, FRDM-K22F, and the FRDM-K64F, which are typical boards for the KL25Z (Core M0+), K22F (Core M4F) and K64F (Core M4F), respectively.

There are a number of existing and future sensor shield boards that can be used in conjunction with the NXP Freedom boards including:

- FRDM-FXS-MULT2-B, an NXP Freedom Development board with Bluetooth containing a variety of NXP sensors including:

  - MMA8652FCFC 3-axis accelerometer
  - FXLS8471Q 3-axis accelerometer
  - MAG3110 3-axis magnetometer
  - FXAS21002C 3-axis gyroscope
  - MPL3115A2 pressure sensor
  - FXOS8700C 3-axis accelerometer plus a 3-axis magnetometer
  - MMA9553L, pedometer sensing platform

- FRDM-FXS-MULTI-B, an NXP Freedom Development board with Bluetooth containing a variety of NXP sensors and connectors for assembling with a Freedom board.
- FRDM-FXS-MULTI, an NXP Freedom Development board containing a variety of NXP sensors and connectors for assembling with a Freedom board.
- FRDM-STBC-AGM01, an NXP Freedom Development board containing the NXP 3-axis gyroscope (FXAS21002C), and the NXP 6-axis Sensor (FXOS8700C). Various kits are available providing both the Freedom board and the shield board.
- FRDM-STBC-SA9500, an NXP Freedom Development board containing the NXP FXLC95000CL 3-axis accelerometer and 32-bit MCU. There is a kitted version providing the FRDM-KL25Z with this development board.
- FRDMSTBC-A8471, an NXP Freedom Development board containing the NXP FXLS8471 3-axis accelerometer. There is a kitted version providing the FRDM-KL25Z with this development board.
- FRDMSTBC-A8491, an NXP Freedom Development board containing the NXP MMA8491Q 3-axis accelerometer. There is a kitted version providing the FRDM-KL25Z with this development board.
- FRDMSTBC-P3115, an NXP Freedom Development board containing the NXP MPL3115A2 pressure sensor. There is a kitted version providing the FRDM-KL25Z with this development board.

The NXP sensors supported in the release according to sensor type are:

Sensor Type	Orderable part numbers
Accelerometer	• FXLC95000CL • FXLN83xx (analog) • FXLS8471Q • FXOS8700C • MMA8491Q • MMA865x (MMA8652FC and MMA8653FC) • MMA9553L • MMA9555L
Analog Sensors	• FXLN83xx (accelerometer)
E-Compass	• FXOS8700C
Magnetometer	• FXOS8700C • MAG3110
Gyroscope	• FXAS21002C
Pedometer	• MMA9553L • MMA9555L
Pressure	• MPL3115A2
Orientation	Sensor Fusion (any combination of accelerometer, gyroscope, and magnetometer)
Intelligent Sensors	• FXLC95000CL • MMA9553L • MMA9555L

3 Release content

Table 2 provides a summary of the contents included on the ISF website located at nxp.com/ISF-2.2-KINETIS.

Table 2. ISF downloads

Run time software	Status
Middleware-Framework	
ISF2P2_PEx.Peupd The ISF v2.2 Processor Expert component library.	Updated
ISF2P2_Installer Installer for ISF v2.2 documentation, tools, and example project files. There is an optional uninstall feature included with the installer, located in the Start menu with the other ISF 2.2 artifacts.	Updated
Application-specific apps, also known as example applications	
ISF2P2_K22F_MQX_PROJ Basic KDS Project using ISF 2.2 with a K22F MCU and MQX. Register Level Interface capability is included.	Unchanged
ISF2P2_K22F_MQX_SF_PROJ KDS Project using ISF 2.2 with a K22F MCU running Sensor Fusion on MQX™	Unchanged
ISF2P2_K64F_FREERTOS_PROJ Basic KDS Project using ISF 2.2 with a K64F MCU and FreeRTOS. Register Level Interface capability is included.	Updated
ISF2P2_K64F_FREERTOS_SF_PROJ Basic KDS Project using ISF 2.2 with a K64F MCU running Sensor Fusion on FreeRTOS	Unchanged
ISF2P2_K64F_MQX_PROJ Basic KDS Project using ISF 2.2 with a K64F MCU and MQX. Register Level Interface capability is included.	Updated
ISF2P2_K64F_MQX_SF_PROJ Basic KDS Project using ISF 2.2 with a K64F MCU running Sensor Fusion on MQX.	Updated
ISF2P2_KL25Z_FREERTOS_PROJ Basic KDS Project using ISF 2.2 with a KL25Z MCU and FreeRTOS. Register Level Interface capability is included.	Unchanged
ISF2P2_KL25Z_MQX_PROJ Basic KDS Project using ISF 2.2 with a KL25Z MCU and MQX. Register Level Interface capability is included.	Unchanged

The ISF Processor Expert component is contained within the *ISF2P2_PEx.PEupd* file loaded from the ISF website, http://www.nxp.com/products/sensors/intelligent-sensing-framework/isf-v2.2-kinetis:ISF-2.2-KINETIS, into Processor Expert via KDS. The component library contains the components listed in Table 3.

Table 3. ISF component status

Explicit Processor Expert Components	Status
ISF_KSDK_Core	Updated
ISF_KSDK_BasicApp	Unchanged
ISF_KSDK_EmbApp	Unchanged
ISF_KSDK_RLI (Register Level Interface)	Updated
Implicit Processor Expert components contained within the updated ISF_KSDK_Core component	**Status**
ISF_KSDK_CommChannel_I2C	Unchanged
ISF_KSDK_CommChannel_SPI	Updated
ISF_KSDK_CommChannel_UART	Updated
ISF_KSDK_Sensor_Generic_Analog	Unchanged
ISF_KSDK_Sensor_Custom_Adapter	Unchanged
ISF_KSDK_Sensor_FXAS21002_Gyrometer	Unchanged
ISF_KSDK_Sensor_FXLC95000_Accelerometer	Updated
ISF_KSDK_Sensor_FXLS8471_Accelerometer	Updated
ISF_KSDK_Sensor_FXOS8700_ECompass	Updated
ISF_KSDK_Sensor_MAG3110_Magnetometer	Unchanged
ISF_ KSDK_Sensor_MMA8491Q_Accelerometer	Updated
ISF_ KSDK_Sensor_MMA865x_Accelerometer supporting MMA8652 and MMA8653	Unchanged
ISF_ KSDK_Sensor_MMA955x_Pedometer supporting MMA9553L and MMA9555L	Unchanged
ISF_ KSDK_Sensor_MPL3115_Pressure	Unchanged
ISF_ KSDK_Sensor_Virtual_Orientation	Updated
ISF_KSDK_BusManager	Unchanged

In addition, the ISF v2.2 installer includes the following documentation, which is also available individually from the Documentation tab of the Documentation page of the ISF 2.2 website.

Table 4. ISF documentation status for ISF v2.2

Documentation	Status
ISF2P2_KINETIS_SW_REFERENCE_RM Software Reference Manual for the Intelligent Sensing Framework v2.2	Unchanged
ISF2P2_KINETIS_API_REFERENCE_RM API Reference Manual for the Intelligent Sensing Framework (Unzip and open ISF_2.2_API_Reference.html)	Updated
ISF2P2_KINETIS_RELEASE_RN Software Release Notes for the Intelligent Sensing Framework v2.2	Updated
ISF2P2_KINETIS_USER_GUIDE_UG User Guide for the Intelligent Sensing Framework v2.2	Unchanged

Furthermore, there are ISF v2.2 training videos available from the Training page of the ISF 2.2 website.

Table 5. ISF training resources status

Training Resources	Status
ISF2.2_Hardware and Software Setup (Training Video showing how to get started with ISF 2.2)	Unchanged
ISF2.2 Installation	Unchanged
ISF2.2 FRDM-K64F Example Project	Unchanged
ISF 2.2 KIT v2 Tool Usage	Unchanged
ISF 2.2 Adding Your Code to the Sample Application Provided	Unchanged
ISF 2.2 Sensor Component Register Settings	Unchanged
ISF 2.2 Creating Host Interface Commands	Unchanged
ISF 2.2 Using the Register Level Interface	Unchanged
ISF 2.2 Orientation Sensor Demonstration	Unchanged

4 What is new for ISF v2.2 (October 2016) release

This release of ISF is a defect fix release of the ISF v2.2 November 2015 baseline. The features of ISF v2.2 are identical to the previous release in November 2015.

4.1 Slave Select Command Structure – ISF v2.2 - October 2016

ISF 2.2 October 2016 introduced a modification to the ISF 2.2 March 2016Slave Select command. In ISF 2.2 March 2016, the MS byte could only be set to 0x00 and the LS byte contained the I^2C Slave Address. In ISF 2.2 October 2016, the Slave Select command is interpreted as follows:

Table 6. Most significant byte of the Slave Select command

Bit	7	6	5	4	3	2	1	0
Symbol	Type	Channel Number (I^2C/SPI)[3:0]			SPI Type (SPI)/Reserved (I^2C)[4:0]			
Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7. Most significant byte of the Slave Select command description

Bit	Symbol	Description
7	**Type**	**0 —** I^2C **1 —** SPI
6 to 4	**Channel Number (I^2C/SPI)[6:4]**	**0 —** I^2C0 / SPI0 (if present else defaults) **1 —** I^2C1 / SPI1 **2 —** I^2C2 / SPI2…
3 to 0	**SPI Type (SPI)/Reserved (I^2C)[3:0]**	**0 —** FXLS8471, FXOS8700 or similar **1 —** FXAS21002 and similar **2 —** MMA955x and similar

Table 8. Least significant byte of the Slave Select command for the type I²C

Bit	7	6	5	4	3	2	1	0
Symbol	Slave ID[7:0]							
Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	7	6	5	4	3	2	1	0
Symbol	Slave Address[7:0]							
Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 9. Least significant byte of the Slave Select command for the type SPI

Bit	7	6	5	4	3	2	1	0
Symbol	Port name, (A = 0, B = 1, C = 2, D = 3, E = 4)[7:5]			Pin number, (0 to 31)[4:0]				
Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 10. Least significant byte of the Slave Select command description

Bit	Symbol	Description
7 to 0	**Slave ID[7:0]**	I²C: Slave address[7:0] SPI: Port name, (A = 0, B = 1, C = 2, D = 3, E = 4)[7:5] Pin number, (0 to 31)[4:0]

Descriptions and Assumptions:

- I²C or SPI: Identifies whether it is an I²C or SPI interface.
- Channel Number: This is the I²C or SPI Channel Number. If more than one I²C or SPI bus is present, this field is used to select the desired bus channel. If only one I²C or SPI bus is present, this field defaults to the one configured. This field is 3 bits wide and can range from 0 to 7.
- SPI Type: The SPI Header can be an R/W field and an address field of 1 or 2 bytes in length. The individual sensor data sheet should be referenced for details regarding SPI Header formats. This command field is 4 bits wide and can accommodate 16 types of SPI devices. The current implementation covers three families of sensors as listed in the initial value table.
- Port Name: This is the name of the MCU port that contains the pin used for the SPI Slave Select to the Sensor. This field is 3 bits wide and can range from 0 to 7 indicating Port Names A to H.
- Pin Number: This is the MCU Pin number within the port used for the SPI Slave Select to the Sensor. This field is 5 bits wide and can range from pin numbers 0 to 31.

- Slave Select Address: This is the I^2C Slave Address. This field is 8 bits wide and can range from 0 to 255

4.2 ISF v2.2 (March 2016) Features

ISF supports all MCUs supported by the Kinetis SDK 1.2. ISF now consists entirely of KSDK PEx Components.

ISF is based on the NXP Operating System (OS) Abstraction Layer. In particular, this is used to support both the MQX™RTOS and FreeRTOS operating systems.

The Sensor Fusion algorithm is version 5.0.

Additional sensors supported include the FXLC95000CL, FXLN83xx, MMA8491Q and MMA955xL.

Sensor adapters using generic APIs based upon sensor types:

Accelerometer
FXLC95000CL
FXLN83xx, analog accelerometer
FXLS8471Q
- I2C support added to existing SPI support

FXOS8700C
- SPI support added to existing I2C support

MMA8491
MMA865x (MMA8652 and MMA8653)
MMA9553L/MMA9555L
- X, Y, Z accelerometer data

Analog Output Sensors (except 5V which is supported by ISF v2.1)
FXLN83xx, analog accelerometer

eCompass
FXOS8700C
- SPI support added to existing I2C support

Gyroscope
FXAS21002C
- SPI support added to existing I2C support

Magnetometer
FXOS8700C
- SPI support added to existing I2C support

MAG3110

Pedometer
MMA9553L/MMA9555L

Pressure
MPL3115

Orientation (Virtual)
Sensor Fusion

4.3 ISF v2.1 – November 2015

All ISF 2.1 functionality is now supported in ISF 2.2, to support Kinetis MCUs, other than the Kinetis E-Series. Most NXP motion sensors are now supported by ISF v2.2. For additional information, see the ISF 2.2 Release Notes on the nxp.com/ISF-2.2-KINETIS website.

CodeWarrior support by ISF has been eliminated in this release. Only Kinetis Design Studio (KDS) is used in the creation and support of this release.

ISF v2.1 has been modified to support devices with analog outputs and are supported only by Kinetis E-Series MCUs. This means that only UART communication protocol is provided, the I^2C and SPI communication protocols have been eliminated.

RLI feature has been removed.

The sensor adapter list has been reduced to Sensor Adapters using generic APIs based upon sensor types:

- Sensors with 5 V analog output
  — MPXV5004DP, differential pressure sensor

All previous sensor adapters have been removed. Users are expected to use ISF 2.2 for all other sensors.

Accelerometer

- FXLC95000CL
- FXLN83xx, analog accelerometer
- FXLS8471Q
  - I2C support added to existing SPI support
- FXOS8700C
  - SPI support added to existing I2C support
- MMA8491
- MMA865x (MMA8652 and MMA8653)
- MMA9553L/MMA9555L
  - X, Y, Z accelerometer data

Analog Output Sensors (except 5 V which is supported by ISF v2.1)

- FXLN83xx, analog accelerometer

eCompass

- FXOS8700C
  - SPI support added to existing I2C support

Gyroscope

- FXAS21002C
  - SPI support added to existing I2C support

Magnetometer

- FXOS8700C
  - SPI support added to existing I2C support
- MAG3110

Pedometer

- MMA9553L/MMA9555L

Pressure

- MPL3115

Orientation (Virtual)

- Sensor Fusion

4.4 ISF v2.1 features

The ISF v2.1 core libraries have been replaced with public source code, in addition to some ease-of-use items.

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 NXP 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 NXP 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 an available and configured accelerometer, magnetometer, or gyroscope. This is an ISF implementation of the NXP Sensor Fusion library.

Additional sensors supported include the MMA8652FC, MMA8653FC, FXLS8471Q, and MPL3115A2. The updated list of sensor adapters using generic APIs is:

Accelerometer

- FXLS8471
- FXOS8700C
- MMA865x (MMA8652 and MMA8653)

Gyroscope

- FXAS21002C

Magnetometer

- FXOS8700C
- MAG3110

Pressure

- MPL3115

Orientation (Virtual)

- Sensor Fusion

4.5 ISF v2.0 features

The middleware is integrated into the NXP 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 NXP ISF website.

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 on sensor types

Accelerometers

- FXOS8700C
- MMA865x

Gyroscope

- FXAS21002C

Magnetometers

- MAG3110
- FXOS8700C

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

The 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.

4.6 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 http://www.nxp.com/products/sensors/intelligent-sensing-framework/isf-v2.2-kinetis:ISF-2.2-KINETIS. Next, click on the text “Download ISF V2.2 Installer” and installer download starts. When finished, click on the text “Download ISF V2.2 PEx PEupd file” and Processor Expert PEupd file download starts.

The Processor Expert PEupd file is a separate download that may be updated with new sensor adapters between releases. This file can be downloaded from the http://www.nxp.com/products/sensors/intelligent-sensing-framework/isf-v2.2-kinetis:ISF-2.2-KINETIS homepage or the Featured Downloads tab. Once downloaded, install the Processor Expert PEupd file within the Kinetis Design Studio (KDS) as follows:

1. Use the Processor Expert pull down menu, and select Import Component
2. Choose the PEupd package
3. Click the Open button

Refer to the NXP Processor Expert website 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 a starting point for users.

6 Release overview

This is the Intelligent Sensor Framework (ISF) release by NXP Semiconductors. It targets the NXP Kinetis platforms using a PIT timer. ISF is built upon the NXP Operating System Abstraction (OSA) supporting both MQX™ and FreeRTOS operating systems. IDE support is provided by KDS.

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

The ISF release makes source, project, and template files available to users. All of the ISF2P2 projects are provided as user applications, containing sample code intended for users to adapt and modify.

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

7 Memory footprint for target: Kinetis K64F

This section describes the memory footprint for compiled ISF v2.2 projects. These projects were generated using the Processor Expert system. The following table shows those Processor Expert components and the final resulting memory size, but it is important to understand that the actual code generated is typically a series of files contained in the project directories. The exact mapping of Processor Expert components to source files is not provided.

Table 11. ISF memory requirements by component type

Component	Code size (bytes)				Description
	Flash		**SRAM**		
	DEC	**HEX**	**DEC**	**HEX**	
Required System Components					
MQX RTOS[1]	52577	CD7F	1420	58C	MQX Kernel
FreeRTOS[1]	15793	3DB1	1624	658	FreeRTOS Kernel
Processor Expert KSDK components	37221	9165	786	312	KSDK Components used by Processor Expert
Commonly Used ISF Core Features					
ISF_KSDK_Core	2243	8C3	22	16	ISF initialization and configuration functions
ISF_KSDK_EmbApp	646	286	104	68	Component
			1200	4B0	Task Stack Space
ISF_KSDK_BusManager	1516	5EC	107	6B	Component
			1024	400	Task Stack Space
ISF_KSDK_Core (Command Interpreter)	4781	12AD	461	1CD	Component
			1200	4B0	Task Stack Space
Optional ISF Components					
ISF_KSDK_RLI[2]	1448	5A8	802	322	Register Level Interface
ISF_KSDK_EmbApp[3]	646	286	104	68	Component
			1200	4B0	Task Stack Space
ISF_KSDK_BasicApp	438	1B6	0	0	Component
			1200	4B0	Task Stack Space
Protocol adapter components					
ISF_KSDK_CommChannel_I2C	772	304	0	0	I^2C Component
ISF_KSDK_CommChannel_SPI	998	3E6	0	0	SPI Component
ISF_KSDK_CommChannel_UART	919	397	0	0	UART Component
Sensor adapter components					
ISF_KSDK_Sensor_FXAS21002_Gyrometer	1968	7B0	24	18	Gyroscope
ISF_KSDK_Sensor_FXLC95000_Accelerometer	2145	861	133	85	Accelerometer
ISF_KSDK_Sensor_FXOS8700_ECompass	2021	7E5	20	14	Accelerometer and Magnetometer
ISF_KSDK_Sensor_MAG3110_Magnetometer	1568	620	16	10	Magnetometer
ISF_KSDK_Sensor_MMA865x_Accelerometer	1660	67C	17	11	Accelerometer

Memory footprint for target: Kinetis K64F

Component	Code size (bytes)				Description
	Flash		SRAM		
	DEC	HEX	DEC	HEX	
ISF_KSDK_Sensor_MPL3115_Pressure	1986	7C2	22	16	Pressure
ISF_KSDK_Sensor_Virtual_Orientation[3]	2740	AB4	52	34	Orientation Sensor
ISF_KSDK_Sensor_MMA955x_Pedometer	1917	77D	127	7F	Accelerometer
ISF_KSDK_Sensor_MMA8491Q_Accelerometer	1334	536	16	10	Accelerometer
ISF_KSDK_Sensor_FXLS8471_Accelerometer	1812	714	165	A5	Accelerometer
ISF_KSDK_Sensor_Generic_Analog[3]	1063	427	25	19	Analog Sensor
ISF_KSDK_Sensor_Custom	958	3BE	16	10	Custom Sensor
Sensor Fusion Embedded Apps	3254	CB6	416	1A0	Application software to send orientation data to Sensor fusion GUI
ISF_KSDK_Sensor_Virtual_Orientation (Sensor Fusion Algorithm)[3]	20690	50D2	8152	1FD8	Dynamic Memory
			0	0	MagCal (Magnetic Calibration) Stack Space
			2000	7D0	Fusion Stack Space

1. ISF requires configuration of an RTOS, either FreeRTOS or MQX.
2. Related to SSDSW-346 change.
3. Related to SSDSW-382 change.

NOTE: Optimization is set to –Os to optimize for code size.

7.1 Minimum ISF configuration

The following ISF components have memory requirements that constitute the minimum ISF memory configuration size, which totals 54953 bytes:

Table 12. ISF memory requirements by component type—minimum

Component	Code size (bytes)				Description
	Flash		SRAM		
	DEC	HEX	DEC	HEX	
FreeRTOS[1]	15793	3DB1	1624	658	FreeRTOS Kernel
Processor Expert KSDK components	37221	9165	786	312	KSDK Components used by Processor Expert
ISF_KSDK_Core	2243	8C3	22	16	ISF initialization and configuration functions
Total Minimum ISF Configuration Size					
Minimum ISF Configuration[2]	55257	D7D9	2432	980	This is the minimal required configuration of ISF.
Maximum Space Available for Customer	993319	F2827	259712	3F680	This is the maximum memory available to customers for application development.

1. ISF requires configuration of an RTOS, either FreeRTOS or MQX.
2. 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.

7.2 Typical Sensor Fusion configuration with ISF

The following components have memory requirements that constitute the minimum memory configuration required for Sensor Fusion with ISF using FreeRTOS, which totals 90427 bytes:

Table 13. ISF memory requirements by component type—FreeRTOS

Component	Code size (bytes)				Description
	Flash		SRAM		
	DEC	HEX	DEC	HEX	
FreeRTOS[1]	15793	3DB1	1624	658	FreeRTOS Kernel
Processor Expert KSDK components	37221	9165	786	312	KSDK Components used by Processor Expert
ISF_KSDK_Core	2243	8C3	22	16	ISF initialization and configuration functions
ISF_KSDK_BusManager	1516	5EC	107	6B	Component
			1024	400	Task Stack Space
ISF_KSDK_CommChannel_I2C	772	304	0	0	I^2C Component
ISF_KSDK_Sensor_FXAS21002_Gyrometer	1968	7B0	24	18	Gyroscope
ISF_KSDK_Sensor_FXOS8700_ECompass	2021	7E5	20	14	Accelerometer and Magnetometer
ISF_KSDK_Sensor_MPL3115_Pressure	1986	7C2	22	16	Pressure
ISF_KSDK_Sensor_Virtual_Orientation	2740	AB4	52	34	Orientation Sensor
Sensor Fusion Embedded Apps	3254	CB6	416	1A0	Application software to send orientation data to Sensor fusion GUI
ISF_KSDK_Sensor_Virtual_Orientation (Sensor Fusion Algorithm)	20690	50D2	0	0	Dynamic Memory
			2000	7D0	MagCal (Magnetic Calibration) Stack Space
			2000	7D0	Fusion Stack Space
Total Typical ISF with Sensor Fusion Configuration Size					
Typical Sensor Fusion with ISF using FreeRTOS[2]	90204	1605C	8097	1FA1	This is the typical configuration for ISF.
Typical Space Available for Customer when using sensor fusion	958372	E9FA4	254047	3E05F	This is the typical memory available to customers.

1. ISF requires configuration of an RTOS, either FreeRTOS or MQX.
2. 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.

NOTE: Optimization is set to –Os to optimize for code size.

8 CPU load

This CPU Load estimate was run on ISF 2.2 March 2016 release, but is assumed to correspond to the expected CPU loading results for ISF 2.2 October 2016 release.

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 NXP FRDM-K64F, 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:

KDS3.0 and the following ISF components were used:

- *ISF_KSDK_Core*, *ISF_KSDK_Bus Manager* (BM), *ISF_KSDK_CommChannel_I2C* and the *ISF_KSDK_Sensor_FXAS21002_Gyrometer*
- MQX RTOS embedded in KDS 3.2 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, which is used by the BM. Each timer count is 1 µsec, as configured by the BM 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, which creates interrupts at the sample rate requested by the user. The timer restarts at a count of zero, after an interrupt is generated. After the sensor data is retrieved by the user's task, the PIT timer 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
- The Task switches to the user-embedded application after a notification of the availability of sensor data

The test application is generated using the standard *ISF_KSDK_EmbeddedApp PEx* component with an FXAS21002C subscription. The initialization section of the embedded application is modified to configure the application to subscribe to sensor data and then start running. A sample counter iss 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 is created to run at the lowest priority of all other tasks. The task runs as a loop that blocks when receiving the Wait-for-Interrupt (WFI) instruction. This test task unblocks when an interrupt for the TPM timer or the MQXLite systick timer occurs The test task 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 14. 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	766621	383	310	73
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 is that the frame 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 383 µsec. Out of the 383 µsec latency—310 µsec were due to serial communication with the sensor. Therefore, ISF code latency is 73 µsec.

Table 15. Measured CPU Load

Frequency (Hz)	Period (msec)	ISF CPU load (%)	CPU load total (%)
1.56	640	0.011	0.06
6.25	160	0.04	0.24
12.5	80	0.09	0.47
50	20	0.35	1.89
100	10	0.70	3.78
200	5	1.40	7.55
400	2.5	2.81	15.10
800	1.25	5.63	30.21

9 Known issues and limitations

9.1 Compiler/IDE issues

With KDS, there are no known compiler or IDE issues.

9.2 Known software issues

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

9.2.1 SPI Support for Register Level Interface

ISF 2.2 has support for the SPI interface to sensors through the Register Level Interface (RLI) application. The Freedom Sensor Toolbox – Community Edition GUI framework does not yet support this feature. As a result, the SPI with RLI interface is available through the use of programs like Real-Term and Tera-Term (serial port terminal emulators).

9.2.2 Open defects

There are no known open defects for this release.

9.2.3 Closed defects

Table 16. Closed defects

	Ticket Summary	Ticket Submission Date	Closed Date	Priority
SSDSW-457	Repeated "Slave Select" commands fail after multiple instances	29 September 2016	3 October 2016	L3
SSDSW-456	Applications freeze at random when using multiple SPI data streams on a given SPI device.	28 September 2016	3 October 2016	L4
SSDSW-434	I^2C Clock speed is not set correctly by RLI example applications	10 August 2016	26 August 2016	L3
SSDSW-428	Push-Pull mode for I^2C pins is specific to the FRDM-K64F project example.	09 August 2016	26 August 2016	L3
SSDSW-397	The last register address in the FXOS8700 Sensor Adapter register map is not programmed.	01 July 2016	26 August 2016	L3
SSDSW-390	TheFXLC95000 Sensor Adapter does not update the default ODR setting on the device.	08 June 2016	29 August 16	L3
SSDSW-382	Sensor Fusion has incorrect roational data and the magnetic calibrations stops collecting data points over time.	22 April 2016	16 May 2016	L3
SSDSW-380	The MMA8491 Sensor Adapter Shutdown command always returns a failure notice.	06 April 2016	30 June 2016	L3
SSDSW-356	The FXLS8471 Sensor Adapter sensor specific configuration swapped the Portrait/Landscape and Pulse detection description fields.	29 February 2016	31 August 2016	L4
SSDSW-345	The RLI Applications only support I^2C.	19 January 2016	21 June 2016	L3
SSDSW-227	The RLI application is not supported in stand-alone mode.	10 August 2015	23 August 2016	L3
SSDSW-133	The FXOS8700 Sensor Adapter is not current with the usage changes in the Data Sheet	12 May 2015	17 August 2016	L3

9.2.4 Defects closed for previous releases

Table 17. Previously Closed Defects

	Ticket Summary	Ticket Submission Date	Closed Date	Priority
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.	18 November 2014	Ticket has not been closed. It is no longer relevant as CW is no longer supported.	L2
SSDSW-132	Sensor Fusion (Orientation Sensor) file fusion_exec.c includes MKL25Z4.h (line 38). This is specific to KL25Z.	11 May 2015	5 October 2015	L3

	Ticket Summary	Ticket Submission Date	Closed Date	Priority
SSDSW-218	ISF 2.1 has timing problem when using both pressure and temperature with MPL3115A3	27 July 15	16 Noveber 2015	L3
SSDSW-232	The Default KSDK component generation for I^2C pin characteristics are not set to Open Drain. This needs to be explicityly set within a project.	17 August 2015	25 August 2015	L3
SSDSW-258	Hybrid mode for FXOS8700C not handled correctly in the sensor adapters.	4 September 2015	1 November 2015	L3
SSWSW-261	Single invocation of App1_ProcessData in Embedded Apps for multiple Sensors after data has arrived.	6 October 2015	30 October 2015	L3
CR312443	The Device info command returns incorrect values for legacy data fields	7 May 2014	30 September 2014	L4
CR318202	CW 10.6 build fails due to code duplication	12 June 2014	28 September 2014	L5
CR325403	Incomplete Error Handling in the DSA	31 July 2014	1 December 2014	L2
CR339505	Bus Manager appears to run out of tokens in the Newton sensor adapter.	12 November 2014	1 December 2014	L3
CR340065	When the host writes an incorrect sample period data to the MAG3110, the incorrect value is saved in memory even though an error message is sent to the user	17 November 2014	8 December 2014	L3
CR340207 (SSDSW-5)	The installation shows unit test information released in the core library.	18 November 2014	24 March 2015	L3
CR340212	Remove compilation warnings (Warnings generated by MQX remain.)	18 November 2014	4 December 2014	L4
CR340215	Improve the error handling in the Embedded Application goto-state.	18 November 2014	25 November 2014	L3
CR340217	Lack of error handling in Sensor Adapters	18 November 2014	25 November 2014	L3
CR340517	Raw sensor Format Selection fails to produce data	20 November 2014	12 December 2014	L3
CR340662 (SSDSW-6)	In order for the CI Streaming feature to work properly, the user must manually type in the following into the Protocol component methods: ci_stream_init ci_protocol_CB_stream	21 November 2014	14 April 2015	L3
CR341078	Bus Manager depends on local declarations of LDD instead of KSDK interface	25 November 2014	12 December 2014	L3
CR341847	Removed code generating unreachable goto code and when optimized fails to call the device manager	2 December 2014	5 December 2014	L3
CR342062	Untyped Data Structure used in App1_processData() User Callback	3 December 2014	4 December 2014	L3
CR345614	The CI mailbox protocol does not use 2 bytes for offset as specified in the SWRM	15 January 2015	30 January 2015	L3

Known issues and limitations

	Ticket Summary	Ticket Submission Date	Closed Date	Priority
CR345907	If the Component name for the ISFEmbApp component is changed, the compilation fails.	19 January 2015	2 February 2015	L3
CR345910	The FXOS8700CQ sensor adapter does not properly support accelerometer only usage at frequencies other than 400 Hz.	19 January 2015	30 January 2015	L3
CR346036	The Device ID Command does not consistently return the expected 18 bytes of data.	20 January 2015	30 January 2015	L2
SSDSW-2	FXAS21002 Sensor Adapter uses pre-production WHOAMI of D6 rather than production value of D7	10 March 2015	25 March 2015	L5
SSDSW-3	There are remnants of the MAG3110 embedded app appearing in the code generated in App1.c	18 March 2015	2 April 2015	L5
SSDSW-4	Need to identify ISF internal tasks that may use floating point and modify their corresponding MQXLite_task to enable MQX_FLOATING_POINT_TASK property.	21 January 2015	2 April 15	L3
SSDSW-7	Remove the extraneous test (1 != 1) in SmSettingsCompare	22 January 2015	2 April 15	L5
SSDSW-8	Fix typos in the generic sensor type definitions	22 January 2015	16 April 2015	L4
SSDSW-9	The fixed point conversion factors in FXOS8700CQ and FXAS21002C sensor adapters are incorrect. The fixed-point acceleration data has 15 fraction bits rather than 16. It is suspected that a similar situation exists 18 March 2015for the magnetometer and gyroscope data as well.	18 March 2015	16 April 2015	L3
SSDSBOX-30 (SSDSW-4)	Some internal tasks using floating point calculations do not enable MQX_FLOATING_POINT_TASK properly.	21 January 2015	2 April 2015	L3
SSDSBOX-49 (SSDSW-8)	There are numerous typographical errors in the generic sensor type definitions	22 January 2015	16 April 2015	L5
SSDSW-64	Change floating point conversion factors in Horizon (MPL3115) sensor adapter.	17 March 2015	24 March 2015	L3
SSDSW-79	Orientation Sensor does not include pressure data, the 10th axis.	16 March 2015	10 April 2015	L2
SSDSBOX-81	Timestamp resolution has only 5 ms resolution by default and may create duplicate timestamps.	3 February 2015	9 February 2015	L3
SSDSW-97	Cycling the Orientation Sensor through STARTED_SUBSCRIBED to STOPPED_UNSUBSCRIBE and back does not work. The isf_fifo_init routine was called inside the sensor adapter configuration API instead of the initialization API.	26 March 2015	10 April 2015	L3

	Ticket Summary	Ticket Submission Date	Closed Date	Priority
SSDSW-99	The response packet for the Command Interpreter does not match the ISF 2.0 Software Reference Manual.	27 March 2015	27 July 2015	L3
SSDSW-98	The stack size for the main task of ISFEmbApp cannot be modified.	27 March 2015	2 April 2015	L3
SSDSW-101	A request to steam data for the FXAS21002 at 800 Hz results in only 480–500 Hz stream data.	2 April 2015	10 April 2015	L3
SSDSW-104	The FXAS21002 sensor adapter Processor Expert component does not allow the user to set sample rates of either 400 or 800 Hz.	8 April 2015	10 April 2015	L3
SSDSW-105	The default task priorities for ISF were not correct and some could not be changed in the Processor Expert components.	8 April 2015	10 April 2015	L3
SSDSW-106	The sensor adapter conversion tables are swapped for the FXAS21002C and the FXAS21000.	10 April 2015	16 April 2015	L3

10 ISF Software change log

Table 18. ISF version changes

Version 2.2 (October 2016)
Bug fixes
Version 2.2 (March 2016))
KSDK Support and additional sensor support
Version 2.1 (November 2015)
KDS, MPXV5004DP with analog output support, KE06Z,
Version 2.1 (April 2015)
Update the PEupd file

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 IDD and IDDA current measurements

This data is required for releases associated with hardware. As ISF is hardware independent, this data is not provided for this release.

12 References

Resource	Description	Link
FXAS21002C	Product Summary Page	nxp.com/FXAS21002
FXLC95000CL	Product Summary Page	nxp.com/ISF-FXLC95000CL
FXLN83xx	Product Summary Page	nxp.com/FXLN83XXQ
FXLS8471	Product Summary Page	nxp.com/FXLS8471Q
FXOS8700C	Product Summary Page	nxp.com/FXOS8700CQ
MMA8491Q	Product Summary Page	nxp.com/MMA8491Q
MMA865x	Product Summary Pages	nxp.com/MMA8652FC nxp.com/MMA8653FC
MMA9553L	Product Summary Page	nxp.com/PEDOMETER
MMA9555L	Product Summary Page	nxp.com/MMA9555L
MPL3115	Product Summary Page	nxp.com/MPL3115A2
MPXV5004DP	Product Summary Page	nxp.com/MPXV5004DP
Kinetis Design Studio	Downloads	nxp.com/KINETIS-IDS/DOWNLOADS
Kinetis Software Development Kit	Documentation	nxp.com/KINETIS-SDK_DOCS
ISF v2.1 website	Tool Summary Page	nxp.com/ISF-2.1-KINETIS
ISF v2.2 installer	Tool Summary Page	nxp.com/isf-2.2-KINETIS
ISF v2.2 Training video	Tool Summary Page	nxp.com/isf-2.2-KINETIS (see training tab)
ISF v2.2 Release Notes	Tool Summary Page	nxp.com/isf-2.2-KINETIS
ISF2P2 website	Tool Summary Page	nxp.com/isf-2.2-KINETIS
Processor Expert website	Tool Summary Page	nxp.com/PROCESSOREXPERT
Sensor Fusion	Tool Summary Page	nxp.com/SENSORFUSION

1. These products are not available at the time of the ISF v2.2 release.

13 Revision history

Document ID	Release Date	Supercedes
ISF2P2_KINETIS_RELEASE_RN v2.0	20161020	ISF2P2_KINETIS_RELEASE_RN v 1.0
Modifications	• Section 2.3, added “and the generic data logger application” to the end of the sentence. • Tables 2, 3, 4, and 5, updated the Status column • Added Section 4.1, Slave Select Command Structure – ISF v2.2 - October 2016 • Table 11, ISF memory requirements by component type • Table 13, ISF memory requirements by component type—FreeRTOS • Added Section 9.2.1, SPI Support for Register Level Interface	
ISF2P2_KINETIS_RELEASE_RN v 1.0	2/2016	Initial public release

How to Reach Us:

Home Page:
NXP.com

Web Support:
NXP.com/support

Document Number: ISF2P2_KINETIS_RELEASE_RN
Revision 2.0, 10/2016