UM2438

User manual

Predictive maintenance reference kit with sensors and IO-Link capability

Introduction

The STEVAL-BFA001V1B reference kit for condition monitoring and predictive maintenance is based on the STEVAL-IDP005V1 high performance industrial sensor platform.

The compact design is especially suitable for monitoring motors, pumps and fans.

The STEVAL-IDP005V1 embeds high performance ARM® Cortex®-M4 32-bit microcontroller (STM32F469AI), iNEMO 6DoF accelerometer and gyroscope (ISM330DLC), a barometric pressure sensor (LPS22HB), a relative humidity and temperature sensor (HTS221), a digital microphone (MP34DT05-A), EEPROM (M95M01-DF) for data Storage, IO-Link PHY device (L6362A) and power management based on a step-down switching regulator and LDO regulator (L6984 and LDK220).

Figure 1. STEVAL-BFA001V1B

The STSW-BFA001V1 firmware package includes:

- Algorithms for advanced time and frequency domain signal processing and analysis of the embedded 3-axis digital accelerometer with 3 kHz flat bandwidth.
- Environmental (pressure, humidity and temperature) monitoring.
- Audio algorithms for acoustic emission (AE) are also part of the package.

The firmware runs on the high performance STM32F469AI, ARM® Cortex®-M4, 32-bit microcontroller. Sensor data analysis results can be transmitted through one of the two available serial communication channels: IO-Link (IO-Link stack is not included on the FW package) or UART.

The package allows connection with an external STEVAL-IDP004V1 IO-Link master multi-port evaluation board. One of the firmware applications and the STEVAL-IDP005V1-GUI user interface included in the firmware package can be used to display data exchange and sensor data. The data can also be displayed using a common terminal emulator like Tera Term, through the UART communication channel.

The condition monitoring demonstration software in the STSW-BFA001V1 package is preloaded in the STM32 Flash memory. Visit the STSW-BFA001V1 product web page on www.st.com to download the latest versions of the demonstration source code and documentation.

1 Overview

The STEVAL-BFA001V1B reference design kit lets you evaluate embedded vibration, environmental and acoustic algorithms for condition monitoring and predictive maintenance applications. You can also reference our hardware and software designs for your own solutions.

The reference material includes:

- Hardware: Schematic, Gerber, BOM
- Software: applications and demonstrations firmware included in STSW-BFA001V1 and dedicated GUI (STEVAL-IDP005V1-GUI) in the STSW-BFA001V1 utilities folder)

Note: A graphical Getting started guide is also available on www.st.com

1.1 Package components

The STEVAL-BFA001V1B package has all the main components that you need for condition monitoring and predictive maintenance applications through the available FW examples and GUI.

Figure 2. STEVAL-IDP005V1 package contents

The package includes:

- A reference design board (10 x 50 mm) - STEVAL-IDP005V1.
- An adapter for ST-LINK programming and debugging tool - STEVAL-UKI001V1.
- A 0.050” 10-pin flat cable.
- A 4-pole cable with M12 female connector.
- A 4-pole mount M12 connector plug, with male contacts.

Figure 3. **STEVAL-IDP005V1 board - top**

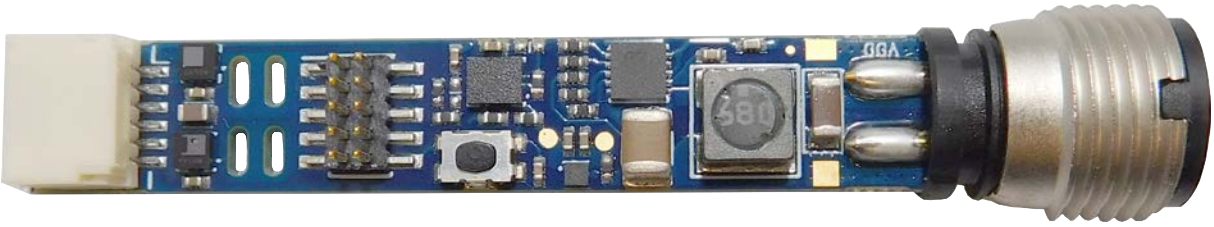
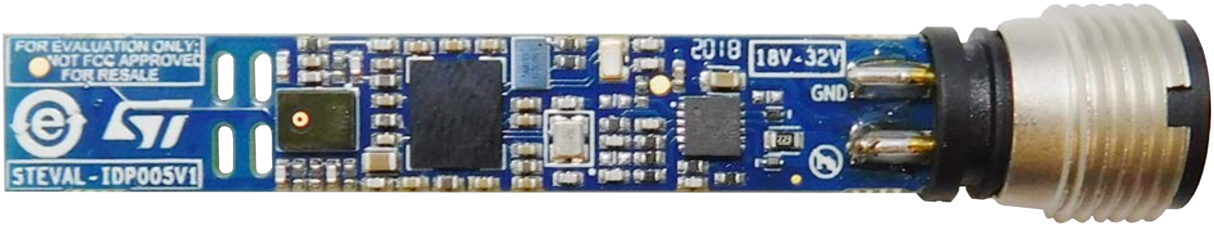


Figure 4. **STEVAL-IDP005V1 board - bottom**



1.2 System requirements

The STEVAL-IDP005V1 is already programmed with Condition Monitoring firmware. To run the demo, you need the following items:

- A Windows™ (version 7or higher) PC with a serial line terminal application like Putty.
- A USB type A to mini B male cable.
- A generic power supply (range 18 to 32 V).
- An STM32 Nucleo 64 board with ST-LINK V2.1 in-circuit debugger/programmer.

To develop your own project, you will also need the following items:

- A Windows™ (version 7or higher) PC with IAR, KEIL or System Workbench for STM32 firmware development environment.
- Microsoft.NET Framework 4.5 or higher (for the GUI only).
- "ST-LINK utility" for binary firmware download (find the latest embedded software version on www.st.com).

1.3 How to run the demo supplied with the firmware

To run the demo, you must first unpack the STEVAL-BFA001V1B kit.

Follow the steps below to run the condition monitoring demonstration firmware (STSW-BFA001V1\Projects \Demonstrations\Condition_Monitoring\CondMonitor_SRV) loaded on the STEVAL-IDP005V1 evaluation board:

Step 1. Plug the STEVAL-UKI001V1 onto the Nucleo board.

Step 2. Connect the STEVAL-UKI001V1 plus Nucleo board assembly to the STEVAL-IDP005V1.

Step 3. Supply power

Step 4. Connect the ST-LINK/V2-1 (on the STM32-NUCLEO 64 board) to the PC through the USB Type-A Male to Type-B mini cable

Step 5. Open and configure your terminal emulator.

Set the following parameters:

- Name: COM Port name
- Baud Rate: 230400
- Data:8
- Parity: None
- Stop Bit: One
- Flow Control: None

Step 6. Push the Reset button on the STEVAL-UKI001V1 (or STEVAL-IDP005V1).

Step 7. Insert the new parameters and/or press ENTER, then press [**Y**] and [**Enter**] to start monitoring.

RELATED LINKS

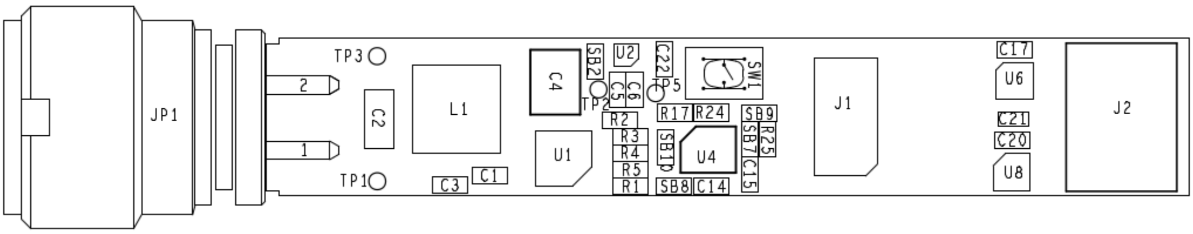
4 How to supply power to the STEVAL-IDP005V1 board on page 16
5.1 Connection through an ST-LINK/V2-1 on page 18
9.1 Outputs for the acoustic analysis project on page 30

2 STEVAL-IDP005V1 hardware architecture

The STEVAL-IDP005V1 has the following main components mounted on the top side:

- JP1 - IO-Link 4-position M12 A-coded connector
- J1 - SWD connector
- J2 - Auxiliary connector
- SW1 - Reset button
- L1 - Shielded power inductor
- U1 - L6984 step-down switching regulator
- U2 - LDK220 LDO
- U4 - ISM330DLC 3D accelerometer and 3D gyroscope
- U6 - HTS221 humidity and temperature sensor
- U8 - LPS22HB pressure sensor

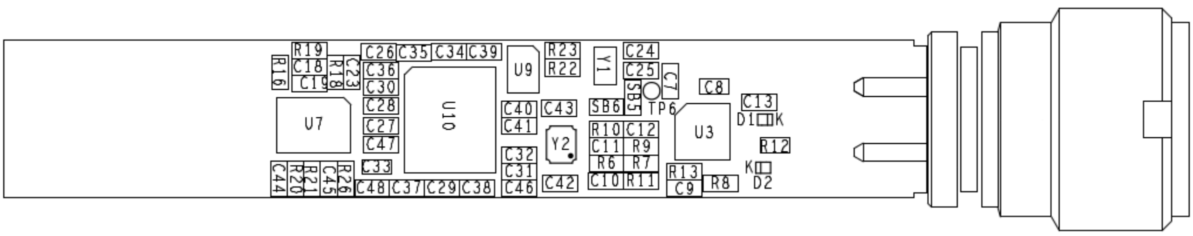
Figure 5. STEVAL-IDP005V1 top side components



The bottom side has the following main components:

- U3 - L6362A IO-Link communication transceiver
- U7 - MP34DT05-A digital microphone
- U9 - M95M01-DF 1-Mbit serial SPI bus EEPROM
- U10 - STM32F469AI ARM® Cortex®-M4 32-bit MCU
- Y1 - 32.768 kHz crystal
- Y2 - 24 MHz crystal

Figure 6. STEVAL-IDP005V1 bottom side components



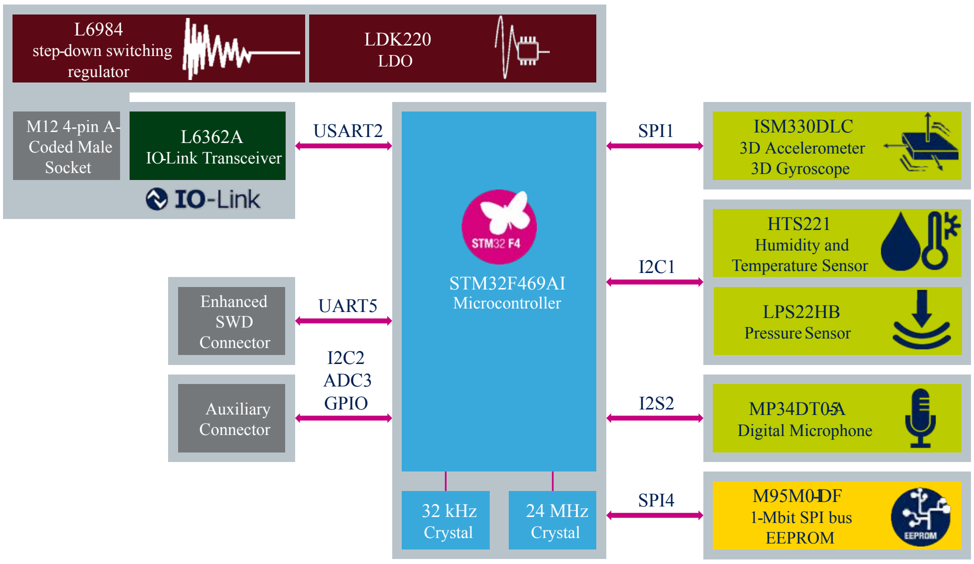
The whole system consists of the following functional subsystems:

1. Power management
2. Microcontroller
3. MEMS sensors
4. EEPROM
5. Wired connectivity
6. External connectors

The sensors are connected to the microcontroller through separate bus SPI and I2C peripherals.

The connectivity options are:
- UART and I2C on the expansion connectors.
- IO-Link on the M12 male socket.

Figure 7. **STEVAL-IDP005V1 functional block diagram**

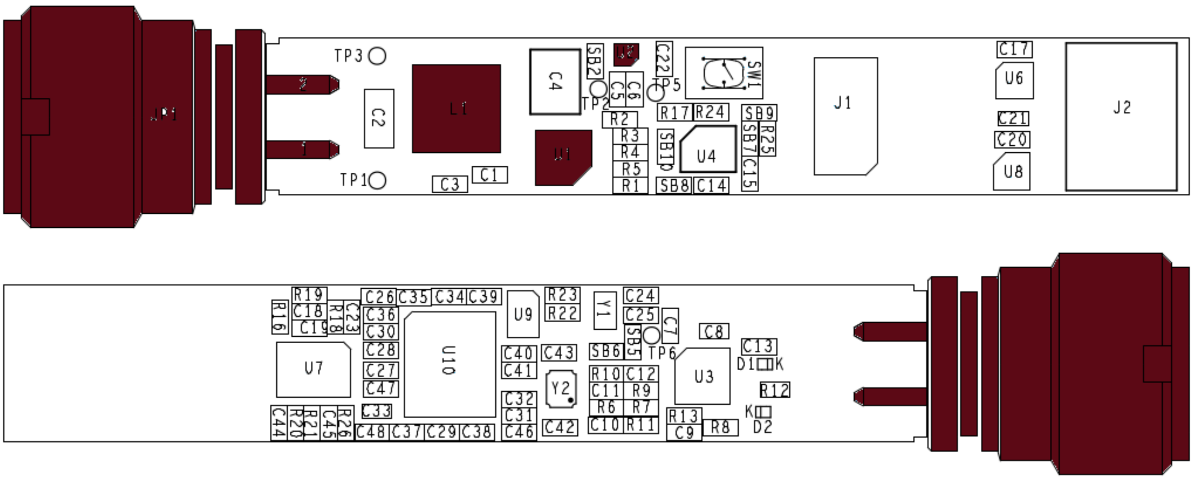


2.1 Power management

The STEVAL-IDP005V1 power management stage can accept an 18 to 32 V_{DC} input through the M12 A-coded 4-pin male connector (JP1) and provide 3.3 V_{DC} / 200 mA voltage output to its digital components.

- U1 - L6984 step-down switching regulator
- U2 - LDK220 LDO

Figure 8. **Power management system**



2.1.1 L6984

The L6984 is a step-down monolithic switching regulator able to deliver up to 400 mA DC. The output voltage adjustability ranges from 0.9 V. The fixed 3.3 V V_{OUT} requires no external resistor divider. The "Low Consumption Mode" (LCM) maximizes the efficiency at light load with controlled output voltage ripple. The "Low Noise Mode" (LNM) makes the switching frequency almost constant over the load current range. The PGOOD open collector output can implement output voltage sequencing during the power-up phase. The synchronous rectification, designed for high efficiency at medium - heavy load, and the high switching frequency capability make the size of the application compact. Pulse-by-pulse current sensing on low-side power element implements an effective constant current protection.

2.1.2 LDK220

The LDK220 is a low drop voltage regulator, which provides a maximum output current of 200 mA from an input voltage in the range of 2.5 V to 13.2 V, with a typical dropout voltage of 100 mV. A ceramic capacitor stabilizes it on the output. The very low drop voltage, low quiescent current and low noise make it suitable for industrial applications. The enable logic control function puts the LDK220 in shutdown mode allowing a total current consumption lower than 1 µA. The device also includes a short-circuit constant current limiting and thermal protection.

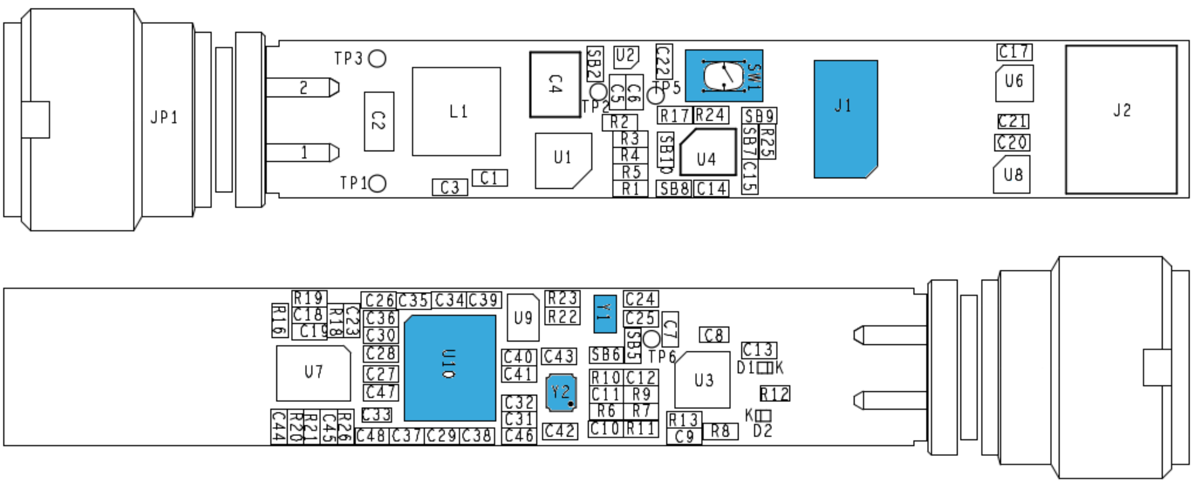
2.2 Microcontroller

The STEVAL-IDP005V1 embeds an STM32F469AI (U10) ARM®Cortex®-M4 32-bit MCU.

The board has a Serial Wire Debug (SWD) connector (J1) for MCU programming and debugging. This connector routes UART pins as well.

The board also has a reset button (SW1) to restart the microcontroller.

Figure 9. Microcontroller subsystem



2.2.1 STM32F469AI

The STM32F469AI microcontroller is based on the high-performance ARM® Cortex®-M4 32-bit RISC core operating at a frequency of up to 180 MHz. The Cortex®-M4 core features a Floating point unit (FPU) single precision which supports all ARM® single-precision data processing instructions and data types. It also implements a full set of DSP instructions and a memory protection unit (MPU) which enhances application security.

The device incorporates high-speed embedded memories (Flash memory up to 2 Mbytes, up to 384 Kbytes of SRAM), up to 4 Kbytes of backup SRAM, and an extensive range of enhanced I/Os and peripherals connected to two APB buses, two AHB buses and a 32-bit multi-AHB bus matrix.

The device offers three 12-bit ADCs, two DACs, a low-power RTC, twelve general-purpose 16-bit timers including two PWM timers for motor control, two general-purpose 32-bit timers, and a true random number generator (RNG).

The microcontroller features the following standard and advanced communication interfaces:

- Up to three I2Cs.
- Six SPIs, two I2Ss full duplex. To achieve audio class accuracy, the I2S peripherals can be clocked via a dedicated internal audio PLL or via an external clock to allow synchronization.
- Four USARTs plus four UARTs.
- One SAI serial audio interface.

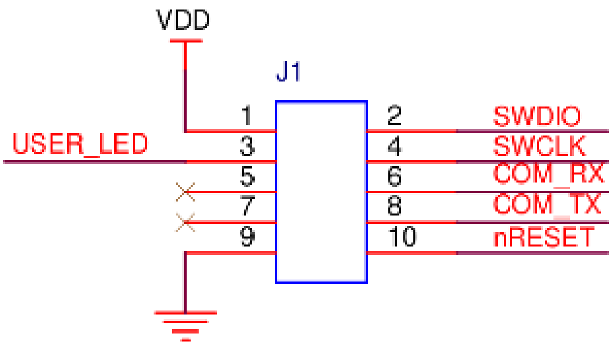
The STM32F469AI device operates in the -40 to +105 °C temperature range from a 1.7 to 3.6 V power supply.

2.2.2 Enhanced SWD connector

The STEVAL-IDP005V1 has a1.27 mm pitch, 10-contact, 2-row board-to-board connector. The connector can be used for the following purposes:

- To program the microcontroller via a dedicated adapter (STEVAL-UKI001V1) connected to the programming tool (e.g. ST-LINK/V2-1).
- As an expansion connector that routes the UART pins, to allow the STEVAL-IDP005V1 to connect with a PC COM port. A further IO for USER_LED is also routed.

Figure 10. Enhanced SWD connector



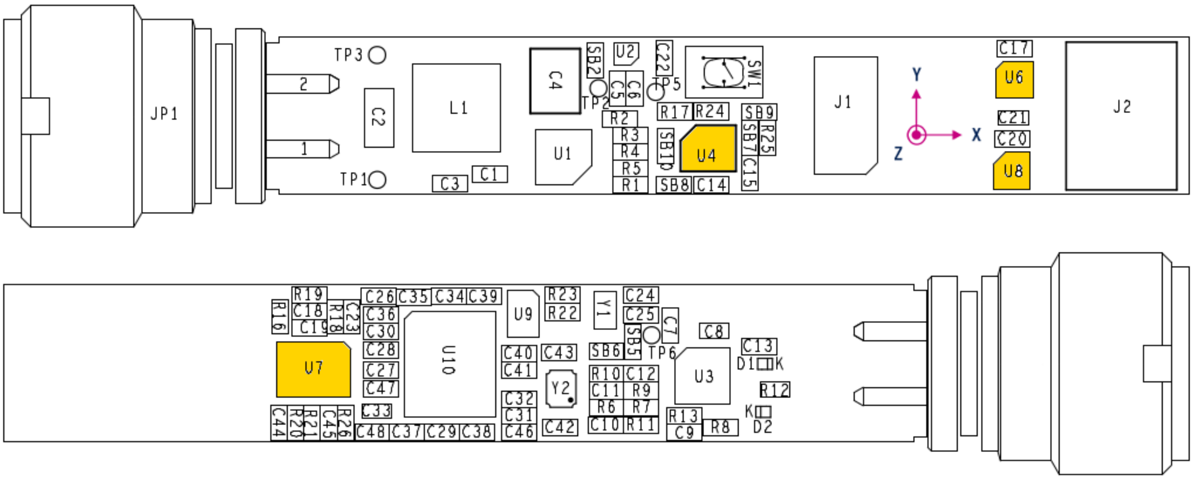
2.3 Sensors

The STEVAL-IDP005V1 embeds several sensors to detect vibration, environmental parameters and sound parameters. The sensor data is analysed with algorithms running on the STM32F469AI microcontroller with FPU.

The following sensors are mounted on the board:

- U4 - ISM330DLC 3D accelerometer and 3D gyroscope
- U6 - HTS221 humidity and temperature sensor
- U8 - LPS22HB pressure sensor
- U7 - MP34DT05-A digital microphone

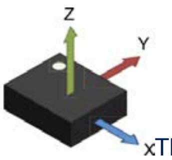
Figure 11. Sensor array subsystem



2.3.1 ISM330DLC

The ISM330DLC is a system-in-package featuring a high performance 3D digital accelerometer and 3D digital gyroscope tailored for Industry 4.0 applications.

ST's family of MEMS sensor modules leverages the robust and mature manufacturing processes already used for the production of micro machined accelerometers and gyroscopes.



The various sensing elements are manufactured using specialized micromachining processes, while the IC interfaces are developed using CMOS technology that allows the design of a dedicated circuit which is trimmed to better match the characteristics of the sensing element.

In the ISM330DLC, the sensing element of the accelerometer and of the gyroscope are implemented on the same silicon die, thus guaranteeing superior stability and robustness.

The ISM330DLC has a full-scale acceleration range of ±2/±4/±8/±16 g and an angular rate range of ±125/±250/±500/±1000/±2000 dps.

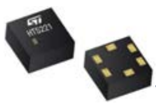
Delivering high accuracy and stability with ultra-low power consumption (0.75 mA in high-performance, combo mode) enables, also in the industrial domain, long-lasting battery operated applications.

The ISM330DLC includes a dedicated configurable signal processing path with low latency, low noise and dedicated filtering specifically intended for control loop stability. Data from this dedicated signal path can be made available through an auxiliary SPI interface, configurable for both the gyroscope and accelerometer. High-performance, high-quality, small size and low power consumption together with high robustness to mechanical shock makes the ISM330DLC the preferred choice of system designers for the creation and manufacturing of versatile and reliable products.

The ISM330DLC is available in a plastic, land grid array (LGA) package.

The STSW-BFA001V1 firmware package includes applications and demonstrations firmware supporting accelerometer part.

2.3.2 HTS221

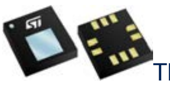


The HTS221 is an ultra-compact sensor for relative humidity and temperature. It includes a sensing element and a mixed signal ASIC to provide the measurement information through digital serial interfaces.

The sensing element consists of a polymer dielectric planar capacitor structure capable of detecting relative humidity variations and is manufactured using a dedicated ST process.

The HTS221 is available in a small top-holed cap land grid array (HLGA) package guaranteed to operate over a temperature range from -40 °C to +120 °C.

2.3.3 LPS22HB



The LPS22HB is an ultra-compact piezo resistive absolute pressure sensor which functions as a digital output barometer. The device comprises a sensing element and an IC interface which communicates through I2C or SPI from the sensing element to the application.

The sensing element, which detects absolute pressure, consists of a suspended membrane manufactured using a dedicated process developed by ST.

The LPS22HB is available in a full-mold, holed LGA package (HLGA). It is guaranteed to operate over a temperature range extending from -40 °C to +85 °C. The package is holed to allow external pressure to reach the sensing element.

2.3.4 MP34DT05-A

The MP34DT05-A is an ultra-compact, low-power, omnidirectional, digital MEMS microphone built with a capacitive sensing element and an IC interface.

The sensing element, capable of detecting acoustic waves, is manufactured using a specialized silicon micromachining process dedicated to producing audio sensors.

The IC interface is manufactured using a CMOS process that allows designing a dedicated circuit able to provide a digital signal externally in PDM format.

The MP34DT05-A is a low-distortion digital microphone with a 64 dB signal-to-noise ratio and -26 dBFS ±3 dB sensitivity.

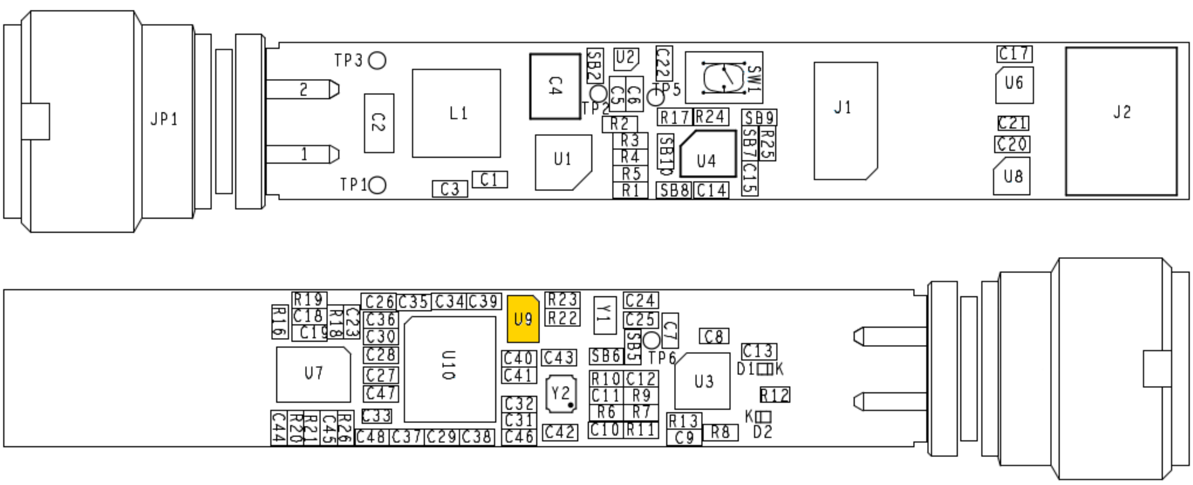
The MP34DT05-A is available in a top-port, SMD-compliant, EMI-shielded package and is guaranteed to operate over an extended temperature range from -40 °C to +85 °C.

2.4 Memory

The STEVAL-IDP005V1 has non-volatile memory which can store up to 1-Mbits of data.

- U9 - M95M01-DF 1-Mbit serial SPI bus EEPROM

Figure 12. EEPROM subsystem



2.4.1 M95M01-DF

The M95M01 electrically erasable programmable memory (EEPROM) id organized as 131072 x 8 bits, accessed through the SPI bus.

The M95M01-DF can operate with a supply range from 1.7 V up to 5.5 V. This device is guaranteed for the -40 °C/+85 °C temperature range.

The M95M01-DF offers an additional Identification Page (256 bytes), which can be used to store sensitive application parameters that can subsequently be permanently locked in Read-only mode.

2.5 IO-Link communication

The STEVAL-IDP005V1 board has IO-Link connectivity available on the M12 A-coded connector.

IO-Link is an industrial standard for hardware connectivity. The standard specifies:

- the number of wires needed for the bus installation
- the colors to distinguish supply voltage from the IO-Link bus line
- connector pinouts.

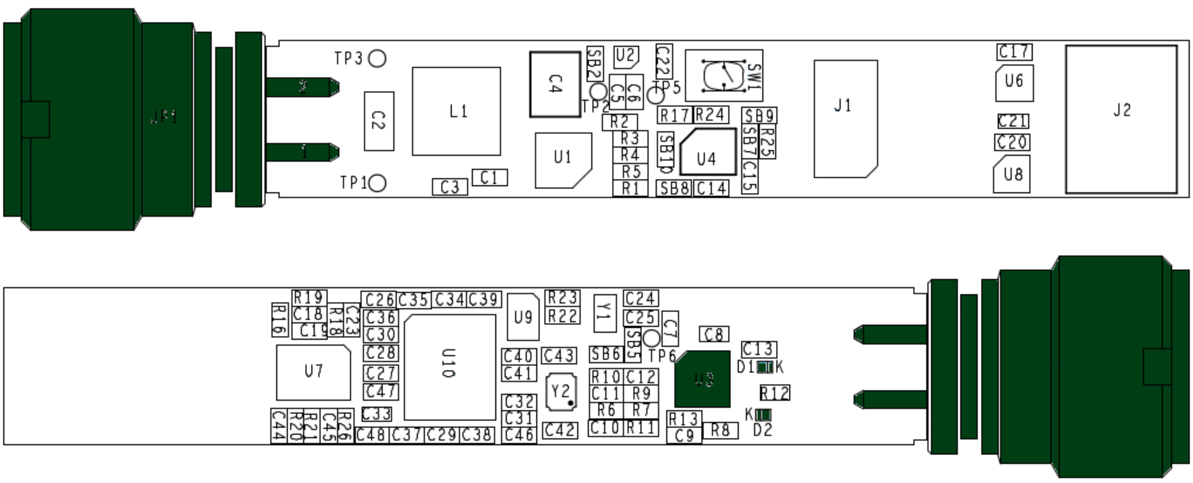
The standard also establishes two different data communication methods:

1. Pure serial data communication (SDCI) with a detailed protocol structure to manage sensor parameters and sensor data.
2. A simple level transition high to low and vice versa to signal the sensor status only.

The use of an IO-Link system offers several advantages, like:

- Automatic detection and parameterization of the IO-Link device: the operating parameters of devices are stored in the master during setup. Once connected, the master recognizes the device and enables automatic startup. If a device like a sensor fails, it can be replaced and parameterization data stored in the master is automatically downloaded to the replacement device.
- Device monitoring and diagnostics: IO-Link allows equipment components and systems to be monitored and proactively managed. Diagnostics provided by IO-Link devices lets the control system track data and trends, facilitating preventive and predictive maintenance and improving machine uptime.
- Changes on the fly: parameters can be quickly adjusted for installed devices while the machine is running, reducing time consumption.
- Reduced component costs: by exploiting the configuration capabilities of IO-Link, a device can be configured to have different output functions.

Figure 13. IO-Link subsystem



2.5.1 L6362A

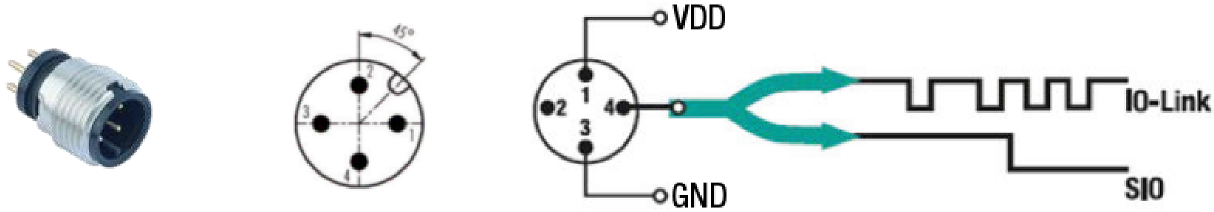
The L6362A is an IO-Link transceiver device compliant with PHY2 (3-wire connection) supporting COM1 (4.8 kbaud), COM2 (38.4 kbaud) and COM3 (230.4 kbaud) modes. The output stage can be configured as high-side, low-side or push-pull by hardware connection, and it can drive resistive, capacitive and inductive loads.

The IC can interface a sensor node to a master unit using both the Serial Data Communication Interface (SDCI) based on IO-Link protocol and the Standard I/O mode (SIO). Communication is managed using the 24 V industrial bus voltage. The L6362A is protected against reverse polarity across VCC, GND, OUTH, OUTL and I/Q pins. The IC is also protected against output short-circuits, overvoltage and fast transient conditions (±1 kV, 500 Ω and 18 μF coupling).

2.5.2 IO-Link connector

The IO-Link connector is M12 A-coded 4-pin.

Figure 14. IO-Link connector and signals



2.6 Auxiliary connections

The STEVAL-IDP005V1 comes with a 6-pin auxiliary connector for:

- V_{DD} and GND
- SMBus (I2C)
- One ADC channel

The above pins can still be used as GPIOs.

The mounted auxiliary connector is a JST SM06B-NSHSS-TB. This mates with a JST NSHR-06V-S, female connector housing, that be assembled with six JST SSHL-003T-P0.2, female crimp terminal contact. These components are not part of the kit.

3 STEVAL-UKI001V1

This tool is an adapter for Serial Wire Debug (SWD) from 10-pin 50-mil socket to 20-pin 100-mil socket (mounted on ST-LINK/V2) or to 6-pin 100-mil (mounted on ST-LINK/V2-1 on the STM32 Nucleo-64 board).

The ST-LINK/V2-1 of the STM32 Nucleo-64 board offers more features. However, you need to ensure that the target application routes the UART RX, UART TX, user button and user LED tracks correctly on the SWD.

You can use ST-LINK/V2-1 through the STEVAL-UKI001V1 board to program and debug the target application. You can also use the ST-LINK/V2-1 as a UART interface adapter via the STM32 Virtual COM Port Driver. This allows you to keep using the USB cable that connects the kit to your PC. To use this configuration, ensure that pins 2 and 3 of CN14 and pins 1 and 2 of CN15 are shorted. Refer to the schematic below.

Figure 15. STEVAL-UKI001V1 schematic

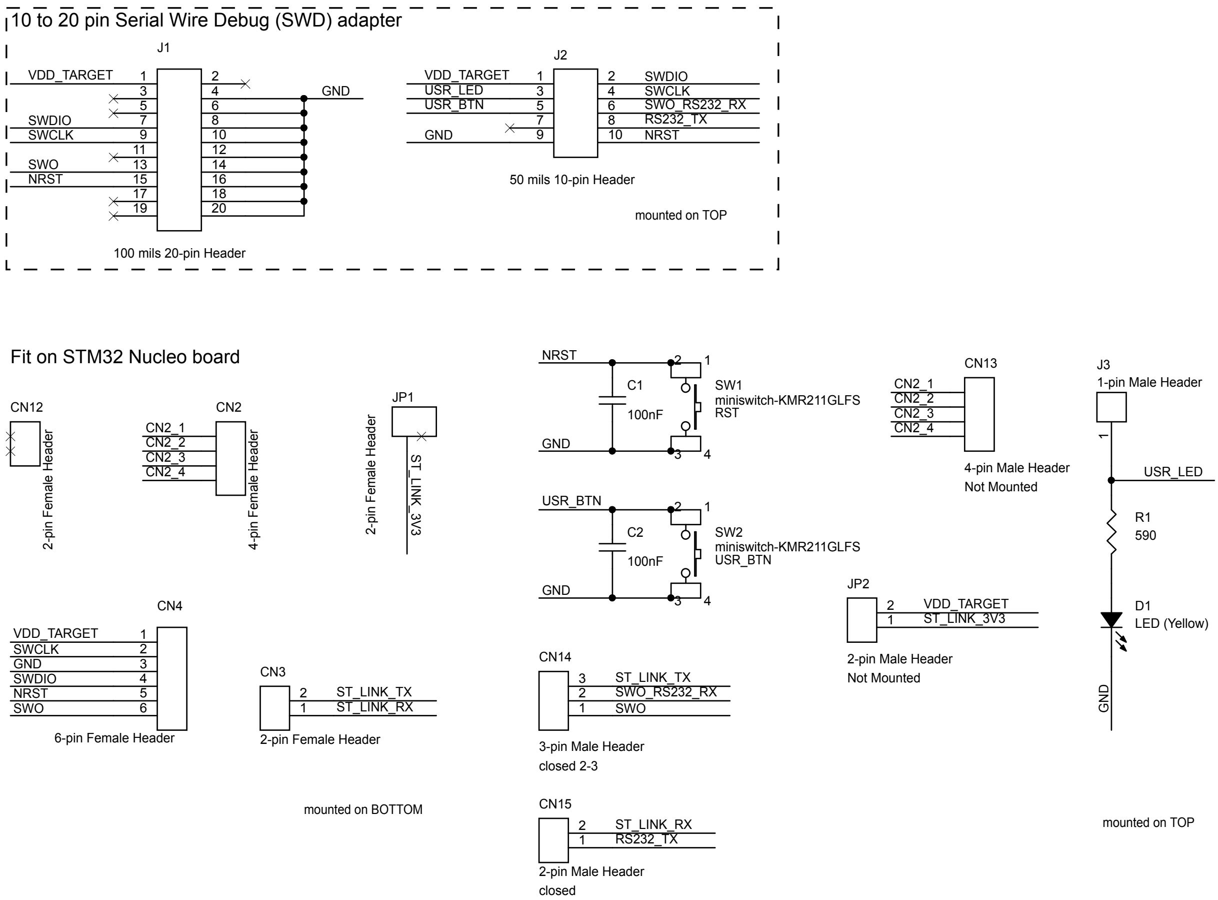


Figure 16. **STEVAL-UKI001V1 top view**

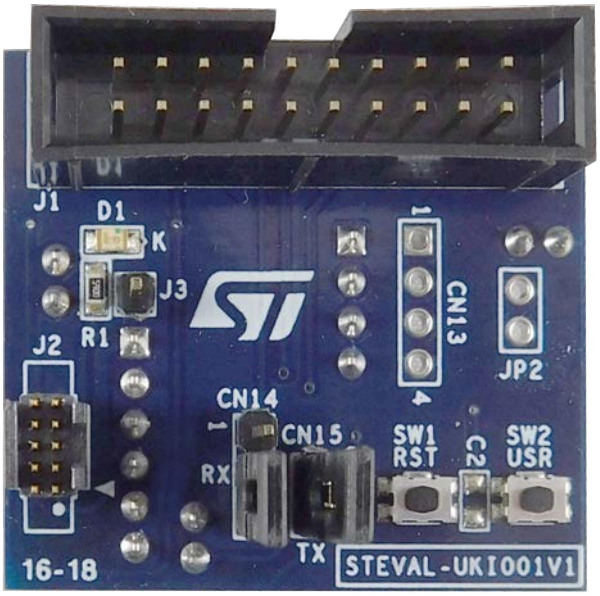


Figure 17. **STEVAL-UKI001V1 bottom view**

4 How to supply power to the STEVAL-IDP005V1 board

The STEVAL-BFA001V1B kit includes the necessary cable and connectors to power the STEVAL-IDP005V1 board.

Figure 18. **4-wire cable with free ends and an M12 A-coded 4-pin female connector**

Figure 19. **4-pole cable mount connector plug with male contacts**

RELATED LINKS

1.3 How to run the demo supplied with the firmware on page 4

4.1 Supply power directly from a DC power supply

You can power the board directly from a DC power supply using only the cable provided in the kit.

Step 1. Connect the cable to an 18 – 32 V_{DC} power supply:
- Pin 1 (brown wire) to positive
- Pin 3 (blue wire) to negative

Figure 20. **STEVAL-IDP005V1 power supply connection (without IO-Link master board)**

4.2 Supply power through an IO-Link master board

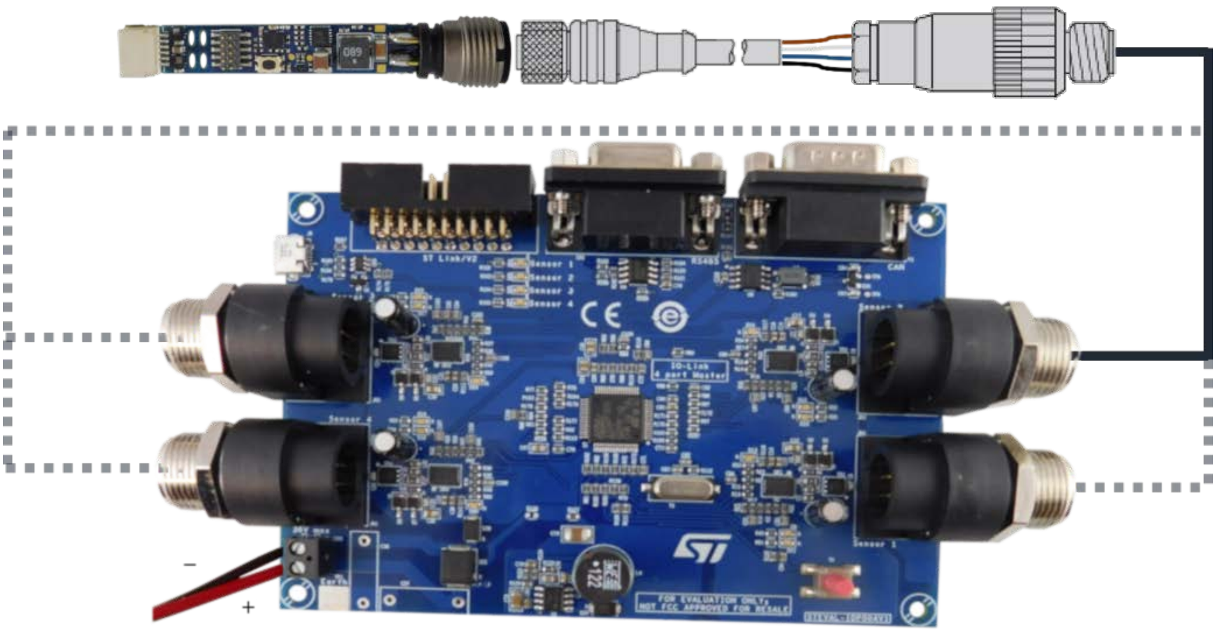
You can supply power via an IO-Link master board using the cable and connectors provided in the kit.

Step 1. Attach the 4-pole cable mount connector plug with male contacts to the cable.

Step 2. Connect the female end to the STEVAL-IDP005V1 board and the male end to the STEVAL-IDP004V1 master board.

Step 3. Power the STEVAL-IDP004V1 IO-Link master board with an 18 to 32 V_{DC} supply through screw connector CON1.

Figure 21. STEVAL-IDP005V1 power supply connection (through IO-Link master board)



5 STEVAL-IDP005V1 board connections

The STEVAL-IDP005V1 needs to be linked with a PC to manage the data coming from the board. The connection can either be through a serial communication adapter (ST-LINK/V2-1) or an IO-Link master multi-port board (STEVAL-IDP004V1).

5.1 Connection through an ST-LINK/V2-1

The ST-LINK/V2-1 in-circuit debugger/programmer on the STM32 Nucleo-64 board lets you update the STEVAL-IDP005V1 firmware. It also allows UART communication with a PC.

To enable UART communication

Step 1. Install the STM32 Virtual COM Port Driver (STSW-STM32102) on your PC.

Step 2. Run a terminal emulator like PuTTY, Tera Term, etc.

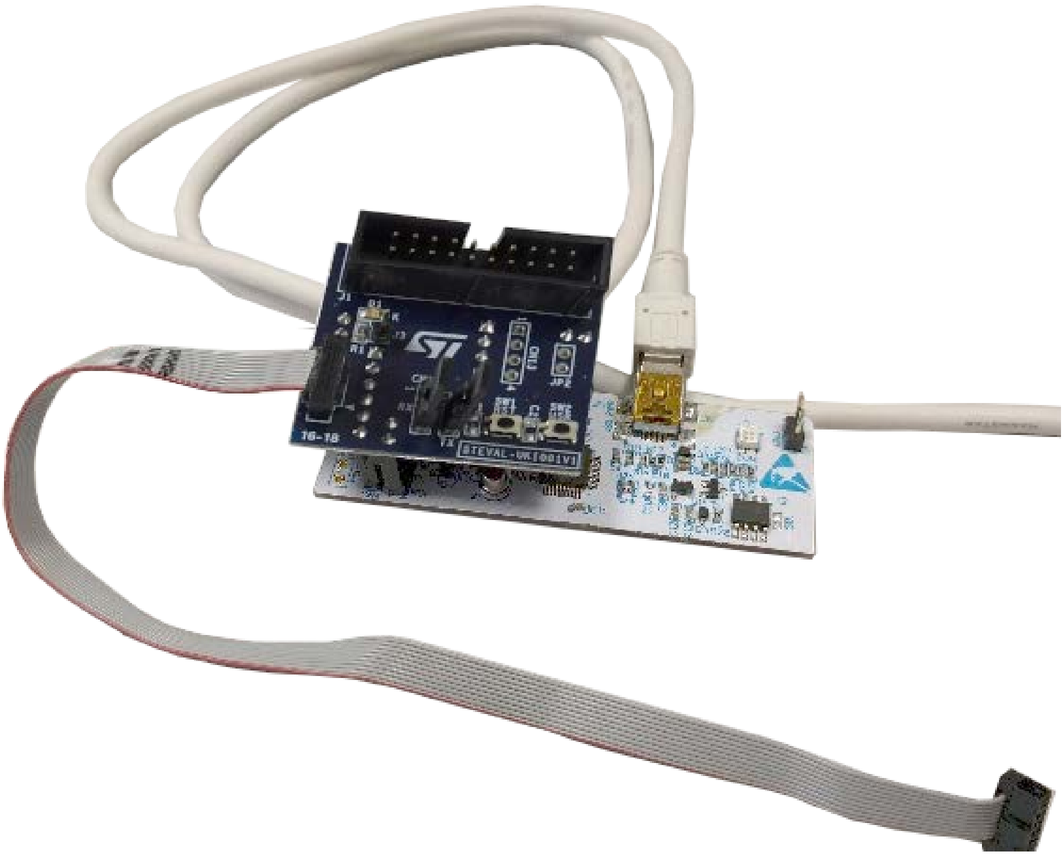
To set up a connection for firmware update.

Step 3. Plug the STEVAL-UKI001V1 on ST-LINK/V2-1 in a manner that the connectors with the same identification are overlapped.

Step 4. Connect the ST-LINK/V2-1 to the PC through the USB Type-A Male to Type-B mini cable.

Step 5. Respecting the polarity, connect an end of the 10-pin flat IDC wire cable to J2 of the STEVAL-UKI001V1.

Figure 22. **ST-LINK/V2-1 connection**



Step 6. On the STEVAL-UKI001V1, short the CN14 pin 2-3 and the CN15.

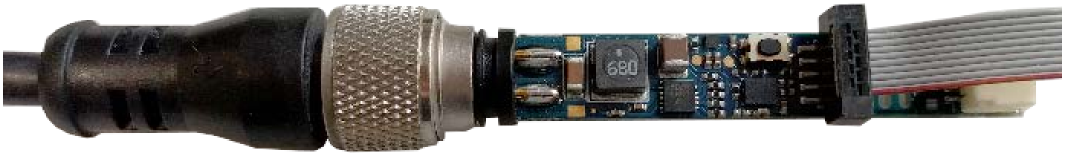
Step 7. Use the 4-wire cable with free ends and an M12 A-coded 4-pin female connector (e.g. Telemecanique Sensors XZCP1141L2).

Step 8. Connect the M12 A-coded 4-pin female connector of the cable to JP1 (IO-Link connector) of the STEVAL-IDP005V1.

Step 9. Connect wire 1 (V_{IN}) and wire 3 (GND) of the cable to a power supply able to provide 18 to 32 V_{DC}.

Step 10. Respecting the polarity, connect the free end of the 10-pin flat IDC wire cable to J1 (SWD connector) of the STEVAL-IDP005V1.

Figure 23. IO-Link and SWD connection



The STEVAL-IDP005V1 is ready to be programmed with new firmware.

RELATED LINKS

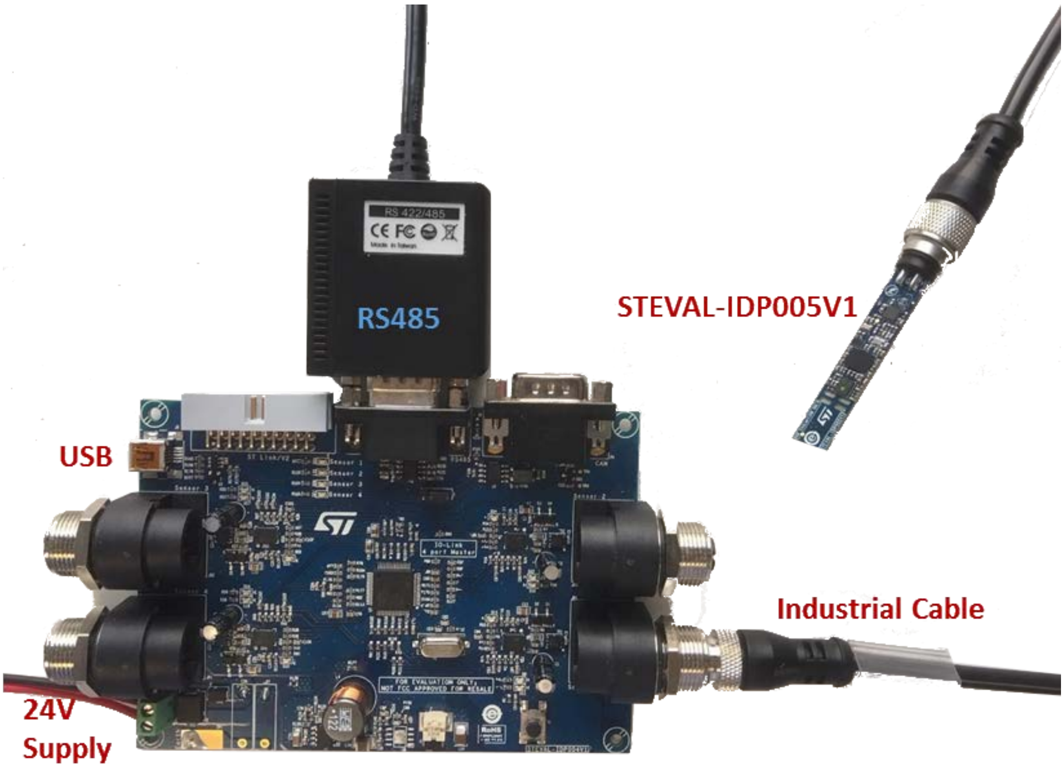
1.3 How to run the demo supplied with the firmware on page 4
8.4.3 Demonstrations folder on page 27
10 How to run projects via IO-Link on page 38
9.1 Outputs for the acoustic analysis project on page 30

5.2 Connection through an STEVAL-IDP004V1

The physical IO-Link connection between STEVAL-IDP005V1 and the PC is made using the STEVAL-IDP004V1 multiport master board with an L6360 master IC for each IO-Link port.

Step 1. Ensure that none of the boards are connected to a power supply.

Figure 24. **STEVAL-IDP004V1 vs STEVAL-IDP005V1 connections**



Step 2. Assemble the Telemecanique Sensors XZCP1141L2 (4-wire cable) with the Telemecanique Sensors XZCC12MDM40B (4-pole connector).

You can also use a preassembled 4-wire cable (not provided in the package) with M12 A-coded 4-pin connectors, male on one end and female on the other.

Step 3. Plug the female M12 connector of the cable to the STEVAL-IDP005V1.

Step 4. Plug the male M12 connector of the cable to a free port of the four ones that are in the STEVAL-IDP004V1.

Step 5. Connect the RS485 dongle (not present in the package) and install the related driver to create the physical connection between PC and master board.

For correct communication, use the reference pinout on the DB9 connector shown below.

Table 1. **RS485 Connector pinout**

PIN Number	PIN Description
1 , 4	Inverting receiver input and inverting driver output
2 , 8	Non inverting receiver input and non-inverting driver output
6 , 7 , 9	Not connected
3 , 5	Ground

Step 6. Connect an 18 to 32 V (typ. 24 V) supply voltage through screw connector CON1 on the board to run the system.

RELATED LINKS

8.4.3 Demonstrations folder on page 27

10 How to run projects via IO-Link on page 38

11 Graphical Interface overview on page 39

6 STEVAL-IDP005V1 firmware overview

The STSW-BFA001V1 software is an expansion of the STM32Cube platform with functions to help you develop applications using inertial, environmental and microphone sensors. The firmware includes sample condition monitoring and predictive maintenance applications based on 3D digital accelerometer, environmental and acoustic MEMS sensors.

The software uses the following lower layers:

- Low level STM32Cube HAL layer to provide all the MCU communication peripherals APIs compatible with STM32Cube framework.
- Low level drivers to facilitate sensor configuration and data reception with dedicated APIs that are compatible with STM32Cube framework.
- Medium level board support package (BSP) layer to provide on-board sensor control and data reception at the application level.

The middleware libraries built on top of the lower layers provide the following features:

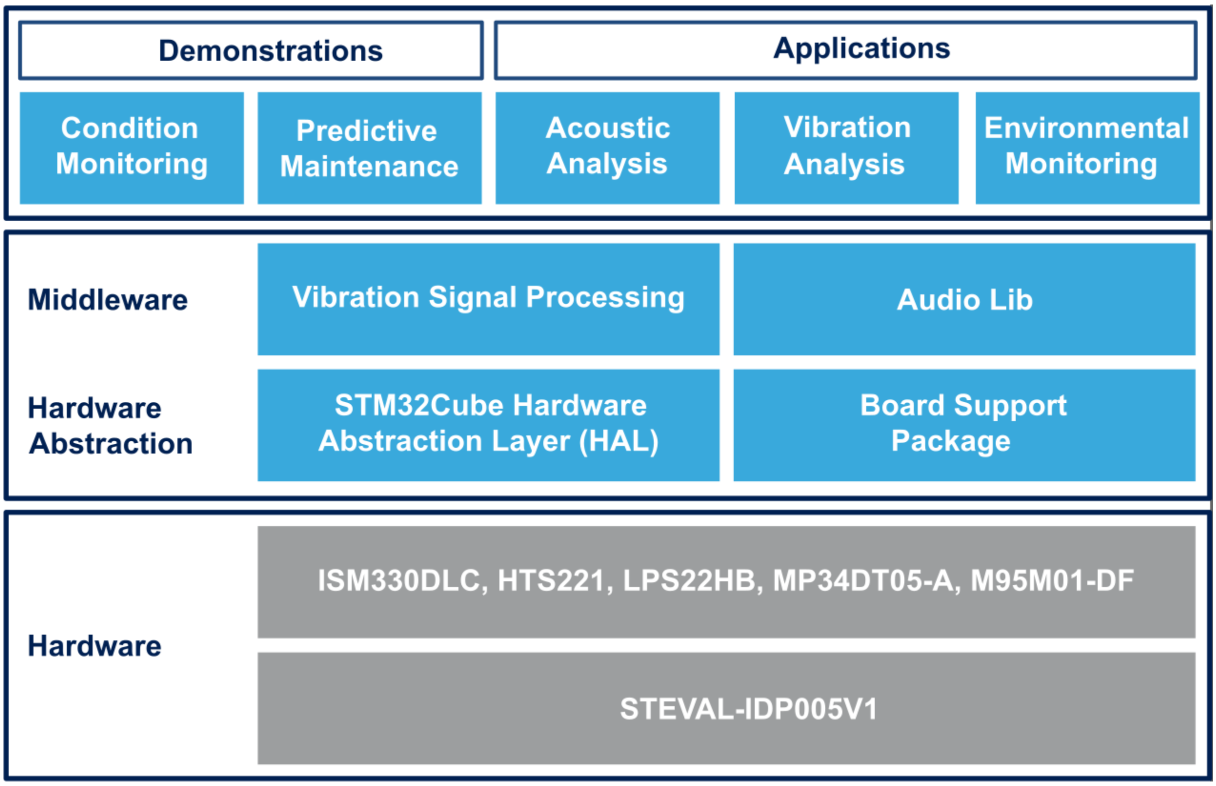
- Middleware, including algorithms for advanced time and frequency domain signal processing for vibration analysis:
  - For the frequency domain:
    - Programmable FFT size (256, 512, 1024, 2048)
    - Programmable FFT input data overlapping
    - Programmable FFT input data windowing (Flat Top, Hanning, Hamming)
    - Programmable FFT output averaging
    - Programmable FFT subrange analysis
  - For the time domain:
    - HP filtering to reduce accelerometer offset
    - Accelerometer max peak evaluation
    - Accelerometer integration to evaluate Speed
    - Moving RMS speed evaluation
- Middleware with microphone algorithms:
  - PDM to PCM
  - Sound pressure
  - Audio FFT
- Sample application to monitor environmental, acoustic and vibration data and read algorithm outputs through a terminal emulator.
- Sample application with programmable warning and alarm thresholds in the time domain and across spectral bands.
- Application example firmware to communicate with STEVAL-IDP004V1 (IO-Link master multi-port evaluation board) and dedicated PC GUI.

7 STEVAL-IDP005V1 firmware architecture

The firmware is based on the STM32Cube™ framework for applications running on the STM32 microcontroller.

The package provides a board support package (BSP) for the MEMS and Microphone sensors and other devices used for IO-Link transmission. The package also contains middleware for signal and audio processing.

Figure 25. STEVAL-IDP005V1 firmware architecture



The following firmware layers access and use the hardware components:

- STM32Cube HAL layer: generic Application Programming Interfaces (APIs) which interact with higher level applications, libraries and stacks. The APIs are based on the common STM32Cube framework so other layers like middleware can function without requiring specific hardware information for a given microcontroller unit (MCU).
- Board support package (BSP) layer: provides firmware support for the STM32 (excluding MCU) peripherals. These APIs provide a programming interface for certain board-specific components like LEDs, user buttons, etc. The APIs can also fetch board serial and version information, as well as support initializing, configuring and reading data from sensors. The BSP provides the drivers for the STEVAL-IDP005V1 board peripherals to connect to the microcontroller peripherals.

This firmware package expands the functionality of the STM32Cube platform with the following features for specific industrial applications:

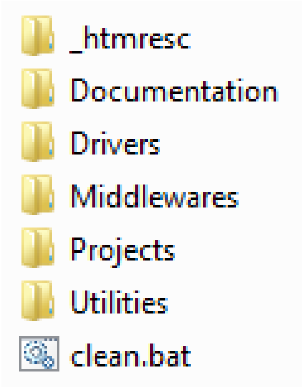
- Low and middle level drivers to connect all the on-board MEMS sensors:
  - Pressure and temperature sensor (LPS22HB)
  - Humidity and temperature sensor (HTS221)
  - Accelerometer/Gyroscope motion sensors (ISM330DLC)
  - Digital Microphone audio sensor (MP34DT05-A)
- Complete BSP functions to allow applications to access sensors. The data acquisition from different sensors is provided via SPI and I2C.
- Six different sample firmware projects divided into two main groups:

  - Applications: examples that use motion, environmental and acoustic measurements, including middleware algorithms focused on vibration and acoustic analysis and environmental monitoring.
  - Demonstrations: projects designed to demonstrate condition monitoring and predictive maintenance with the STEVAL-IDP005V1. The projects include IO-Link connectivity with the master board (STEVAL-IDP004V1).
- Command line interface (CLI) using a debug console on an external terminal via UART communication with a PC.

8 STEVAL-IDP005V1 firmware folder structure

The STSW-BFA001V1 package is developed using the standard STM32Cube™ framework structure shown below.

Figure 26. STSW-BFA001V1 firmware folder structure



8.1 Documentation

The documentation folder contains a compiled HTML file generated from doxygen comments in the source code. The folder also has documentation regarding the firmware framework, drivers for the on-board components and APIs to manage the different functions.

Note: For more information, open the STEVAL-IDP005V1_FW.chm help file in the documentation folder.

8.2 Drivers

All firmware packages compliant with the STM32Cube framework contain the following main groups:

- BSP: board-specific drivers for the HW components.
- CMSIS: vendor-independent hardware abstraction layer for the ARM Cortex-M series, including DSP libraries used for the projects.
- STM32F4xx_HAL_Drivers: microcontroller HAL libraries.

The board support package files are grouped into two main folders with the low level hardware device drivers and the board-specific medium level drivers:

- Components: includes a set of platform-independent device drivers for LPS22HB, HTS221, ISM330DLC, M95M01-DF, as well as common files.
- STEVAL-IDP005V1: includes a set of medium level drivers for each hardware subsystem. You can use the drivers in your application to control and configure the functionality of different measurement datatypes.

These APIs abstract the on-board hardware and connectivity devices contained in the steval_idp005v1 module for use by applications.

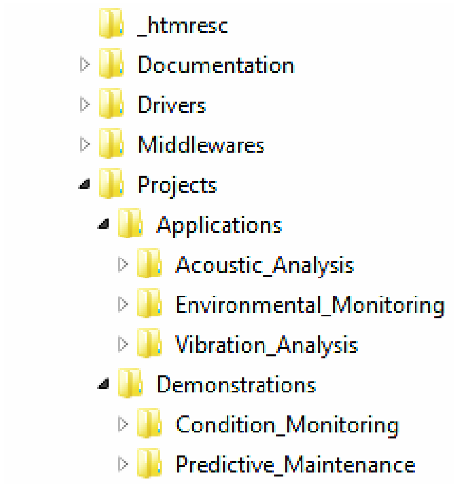
8.3 Middleware

The Middlewares folder contains two specific libraries that give higher level applications access to APIs for acoustic and motion signal processing analysis.

8.4 Projects

The Projects directory contains several user projects under Applications and Demonstrations subfolders.

Figure 27. Projects folders



All the projects are available for the following integrated development environments (IDE):

- IAR Embedded Workbench® for ARM® (EWARM) by IAR systems®
- Microcontroller Development Kit for ARM® (MDK-ARM) by Keil®
- System Workbench for STM32 (SW4STM32) by AC6 (free IDE)

8.4.1 Standard files for all projects

The standard STM32Cube application files have the same configuration as any standard example using the STM32 HAL libraries, plus the peripherals used for demonstration purposes in the following files:

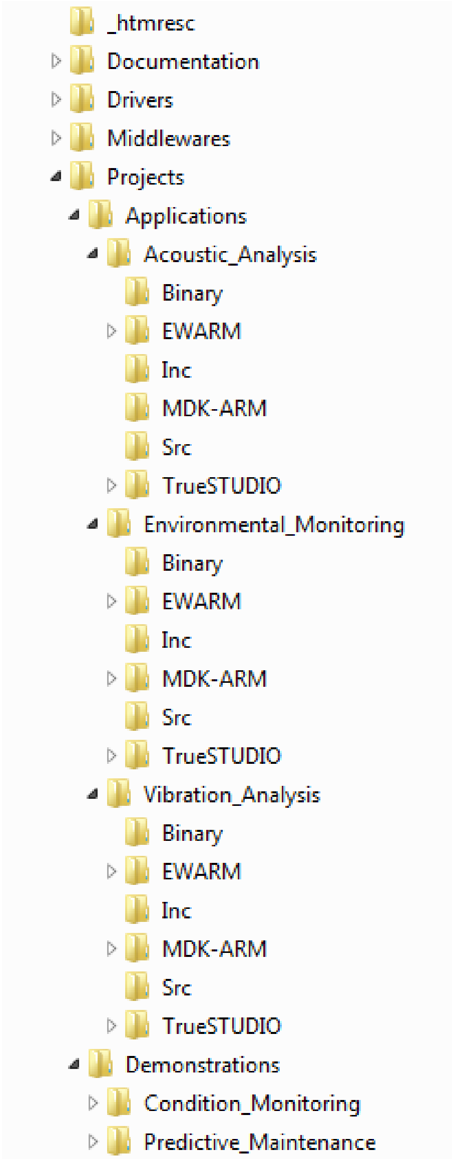
- main.c: APIs for system clock configuration and all the standard include files for the other APIs defined in HAL libraries, BSP and Middleware.
- stm32f4xx_hal_msp.c: APIs for application-level peripheral initialization.
- stm32f4xx_hal_it.c: APIs for all interrupt handlers.

8.4.2 Applications folder

The Applications folder includes separate projects and reference firmware to monitor (through serial communication via the STEVAL-UKI001V1) the following types of data from the STEVAL-IDP005V1:

1. Vibration data: with vibration analysis based on accelerometer data for diagnostic purposes.
2. Audio data: retrieves sound data such as sound pressure level and sound power spectrum.
3. Environmental data: retrieves environmental data such as humidity, temperature and pressure.

Figure 28. **Applications folders**



8.4.2.1 *Application-specific files for projects in the Applications folder*

The application-specific APIs for vibration analysis are found in the following files:

- main.c:
  - APIs for sending application information to a terminal screen (via Service UART)
  - APIs for sensor initialization (accelerometer)
  - APIs for sensor measurement (accelerometer)
  - APIs for accelerometer parameters that can be configured by the user, and accelerometer INT management
  - APIs for time domain and frequency domain analyses
- data_communication_srv.c: APIs for the CLI configuration command and to monitor the processing outputs requested by the user.
- MotionSP_Manager.c: to interface with middleware functionality.

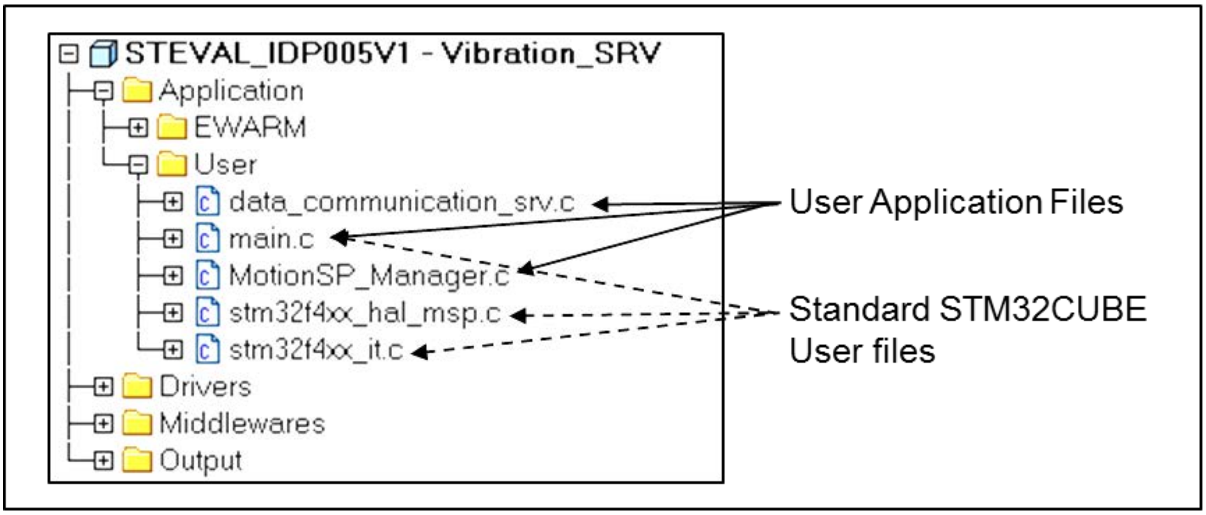


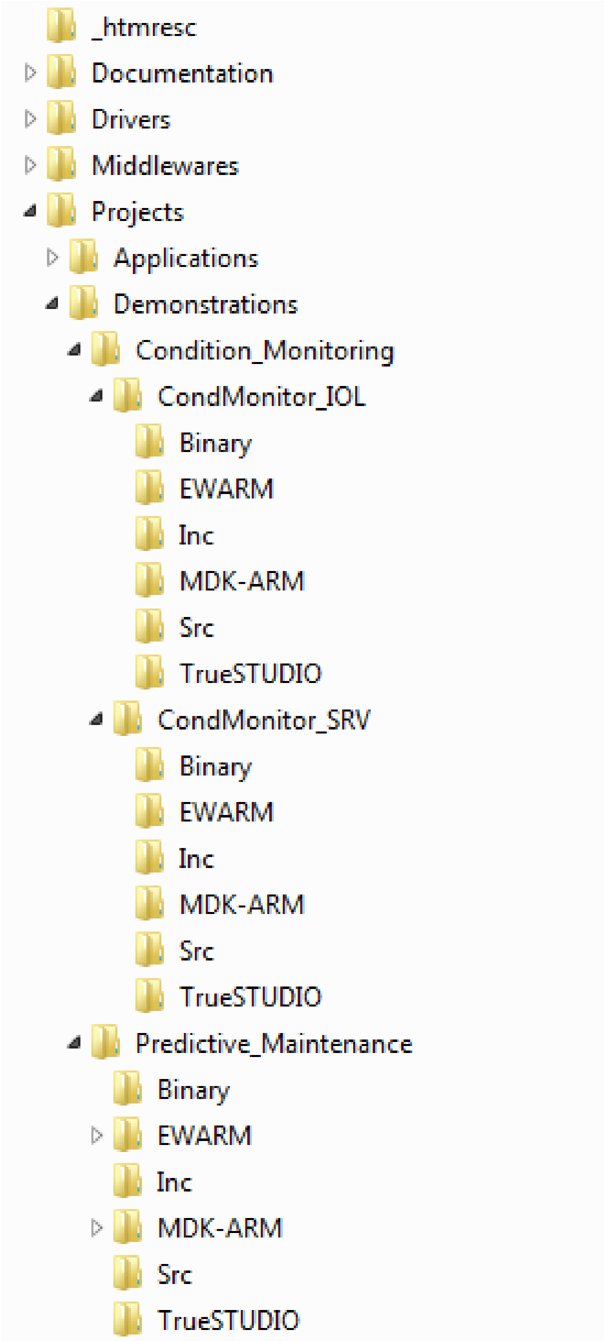
Figure 29. User files for vibration analysis

8.4.3 Demonstrations folder

The Demonstrations folder includes three projects for the STEVAL-IDP005V1:

1. Predictive Maintenance with serial communication via STEVAL-UKI001V1.
2. Condition Monitoring:
   a. with serial communication via IO-Link
   b. with serial communication via UART through the STEVAL-UKI001V1

Figure 30. Demonstrations folder



The project for Predictive Maintenance analyzes vibration data against threshold parameters for the same measurement datatype evaluated. The project includes a algorithm to determine status information with respect to time and frequency domain parameters.

There are two Condition Monitoring projects designed to retrieve and analyze sensor data to evaluate equipment status.

The two projects differ in how the data is transmitted.

1. The Conditon_Monitoring_SRV project uses standard communication with a PC via the STEVAL-UKI001V1 mounted on STLINK/V2-1.
2. The Conditon_Monitoring_IOL project uses the IO-Link communication PHY, interfacing the STEVAL-IDP005V1 with the STEVAL-IDP004V1 master board and sending the received data via a RS485-USB adapter to a PC. This methods lets you monitor the system with the STSW-IO-LINK GUI.

RELATED LINKS

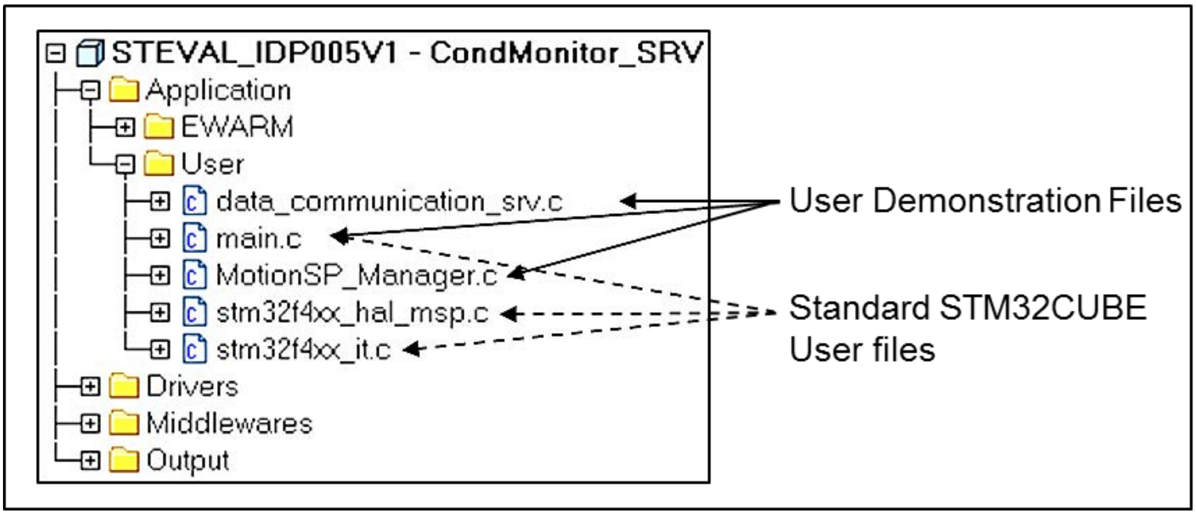
5.1 Connection through an ST-LINK/V2-1 on page 18
5.2 Connection through an STEVAL-IDP004V1 on page 19
11 Graphical Interface overview on page 39

8.4.3.1 *Application-specific files for the Conditon_Monitoring_SRV project*

The application-specific APIs for the Conditon_Monitoring_SRV project are found in the following files:

- main.c
  - APIs for sending the application information to the terminal screen (via Service UART)
  - APIs for sensor initialization (accelerometer, humidity, pressure and temperature)
  - APIs for sensor measurement (accelerometer, humidity, pressure and temperature)
  - APIs for external memory Init (EEPROM)
  - APIs for accelerometer parameters that can be configured by the user, and accelerometer INT management
  - APIs for time domain and frequency domain analyses
- data_communication_srv.c: APIs to run the CLI configuration command and to monitor requested processing outputs.
- MotionSP_Manager.c: to interface with middleware functionality.

Figure 31. **for Conditon_Monitoring_SRV project**



8.4.3.2 *Application-specific files for the Conditon_Monitoring_IOL project*

The application-specific APIs for the Conditon_Monitoring_IOL project are found in the following files:

- main.c
  - APIs for sending the application information to the terminal screen (via IO-Link PHY device)
  - APIs for sensor initialization (accelerometer, humidity, pressure and temperature)
  - APIs for sensor measurement (accelerometer, humidity, pressure and temperature)
  - APIs for time domain and frequency domain analyses
- data_communication_iol.c: APIs designed to receive many customized commands from a board with IO-Link Master, and to send sensor and processing datatypes. Master-slave node communication is managed through the IO-Link channel.
- MotionSP_Manager.c: to interface with middleware functionality.

9 How to run projects via Service UART

Perform the following steps for any of the projects available for Service UART:

Step 1. Connect the STEVAL-IDP005V1 to an ST-LINK/V2-1 in-circuit debugger/programmer on the STM32 Nucleo-64 and download the dedicated firmware.

Step 2. Run a terminal emulator like PuTTY on your PC

Be sure to use the correct COM port and UART parameters.

Step 3. Press the reset button to restart the application.

9.1 Outputs for the acoustic analysis project

The terminal emulator for acoustic analysis shows the following information:

Figure 32. Terminal emulator screenshot for acoustic analysis firmware

```
STEVAL-IDP005V1 [Application - Acoustic - FW v.1.0.0]
2018 STMicroelectronics

MCU ID....: 003600383035511939383238
MCU SYSCLK: 180000000 Hz
MCU HCLK..: 180000000 Hz
MCU PCLK1.: 45000000 Hz
MCU PCLK2.: 90000000 Hz

Audio Sampling Frequency: 48000 Hz
FFT size                : 1024
FFT bin width           : 46.875 Hz

           | Sound Pressure | Sound Power Peak
-----------+----------------+--------------------------
 MP3DT05-A |     62 dB(SPL) |  -57.36 dB @ 20015.625 Hz

Average power spectrum within 5000 ms of time window.
```

The log file from the terminal emulator will store the following information:

- the measured sound pressure and its acquisition time
- the measured average power spectrum with its peak and its acquisition time.

```
=~=~=~=~=~=~=~=~=~=~=~= PuTTY log 2018.05.31 16:51:28 =~=~=~=~=~=~=~=~=~=~=~=
  STEVAL-IDP005V1 [Application - Acoustic - FW v.1.0.0]
 2018 STMicroelectronics
  MCU ID....: 003600383035511939383238
 MCU SYSCLK: 180000000 Hz
 MCU HCLK..: 180000000 H
 MCU PCLK1.: 45000000 Hz
 MCU PCLK2.: 90000000 Hz
  Audio Sampling Frequency: 48000 Hz  FFT size                : 1024  FFT bin widt
h           : 46.875 Hz
  | Sound Pressure | Sound Power Peak
  -----------+----------------+--------------------------
   MP3DT05-A |    --- dB(SPL) | ----.-- dB @ -----.--- Hz
```

```
      1681 ms    87 dB(SPL)       1745 ms    57 dB(SPL)
(omissis)
     6546 ms    63 dB(SPL)       6610 ms    62 dB(SPL)       6633 ms    -62.42 dB @
515.625 Hz
  Average power spectrum within 5000 ms of time window.
     46.875 Hz @  -64.05 dB      93.750 Hz @  -66.98 dB
(omissis)
 23906.250 Hz @  -93.67 dB  23953.125 Hz @  -93.55 dB       6887 ms    61 dB(SPL)
    6930 ms    61 dB(SPL)
(omissis)
```

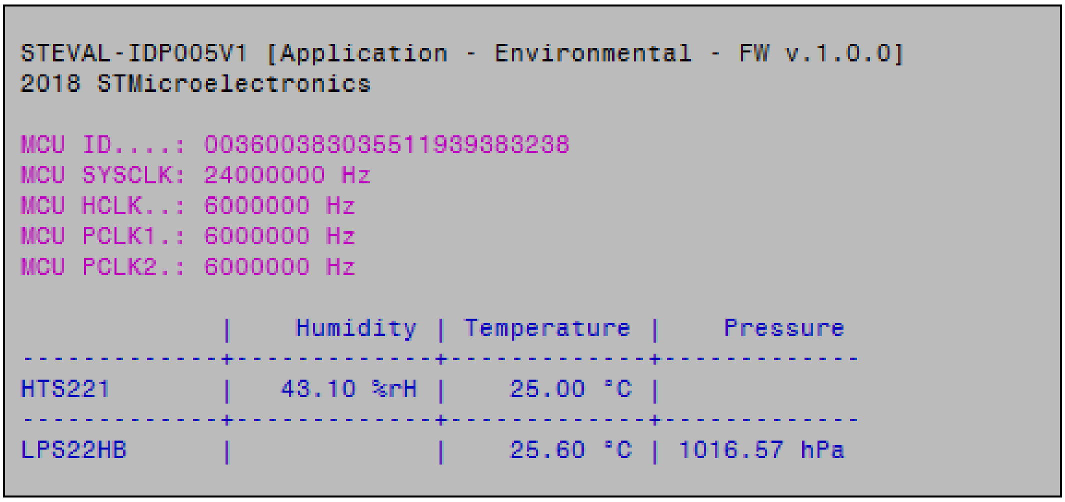
RELATED LINKS

1.3 How to run the demo supplied with the firmware on page 4
5.1 Connection through an ST-LINK/V2-1 on page 18

9.2 Outputs for the environmental monitoring project

The terminal emulator will show the following information:

Figure 33. **Terminal emulator screenshot while running environmental monitoring firmware**



```
STEVAL-IDP005V1 [Application - Environmental - FW v.1.0.0]
2018 STMicroelectronics

MCU ID....: 0036003830355119393832 38
MCU SYSCLK: 24000000 Hz
MCU HCLK..: 6000000 Hz
MCU PCLK1.: 6000000 Hz
MCU PCLK2.: 6000000 Hz

              |    Humidity | Temperature |    Pressure
--------------+-------------+-------------+-------------
HTS221        |   43.10 %rH |    25.00 °C |
--------------+-------------+-------------+-------------
LPS22HB       |             |    25.60 °C | 1016.57 hPa
```

The log file from the terminal emulator will store the following information:

- the measured sound pressure and its acquisition time

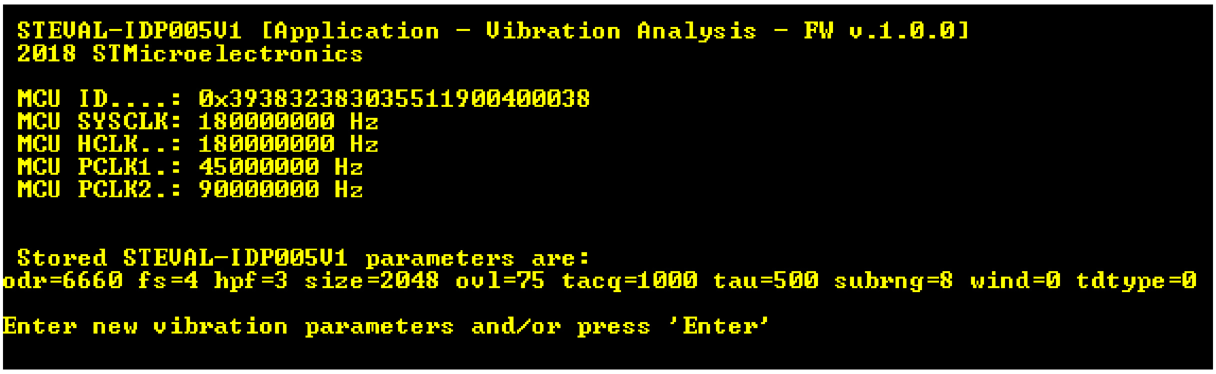
```
=~=~=~=~=~=~=~=~=~=~=~= PuTTY log 2018.05.30 19:23:16 =~=~=~=~=~=~=~=~=~=~=~=
 STEVAL-IDP005V1 [Application - Environmental - FW v.1.0.0]
 2018 STMicroelectronics
 MCU ID....: 003600383035511939383238
 MCU SYSCLK: 24000000 Hz
 MCU HCLK..: 6000000 Hz
 MCU PCLK1.: 6000000 Hz
 MCU PCLK2.: 6000000 Hz
 |    Humidity | Temperature |    Pressure
 -------------+-------------+-------------+-------------
 HTS221       |   ---.-- %rH |   ---.-- °C |
 -------------+-------------+-------------+-------------
 LPS22HB      |
 |   ---.-- °C | ----.-- hPa
 00:00:01 |  HTS221 |  45.80 %rH |  23.80 °C |  LPS22HB |  24.40 °C | 1016.47 hPa
```

```
00:00:06 |  HTS221 |  45.60 %rH |  23.90 °C |  LPS22HB |  24.50 °C | 1016.42 hPa
(omissis)
```

9.3 Outputs for the vibration analysis project

The terminal emulator will show the following information:

Figure 34. Terminal emulator screenshot while running vibration analysis firmware



The bottom part of the screen lists the stored parameters for the analysis and prompts you to change any of these parameters. The configurable parameters are:

odr
Use the same values available for the specific accelerometer (ISM330DLC) to ensure high performance: 13 (for 12.5), 26, 52, 104, 208, 416, 833, 1660, 3330, 6660. See the ISM330DLC datasheet for further details.

fs
The configurable values are: 2, 4 (default), 8, 16. See ISM330DLC datasheet for further details.

hpf
Cutoff frequency for internal hardware High Pass Filter (HPF) as per the following table:

Table 2. HPF configuration

HPF configuration	Cutoff frequency selected
0	ODR/4
1	ODR/100
2	ODR/9
3 (Default)	ODR/400
4	NO_HPF

size
FFT input array accelerometer size: 256, 512, 1024, or 2048 (default)

ovl
Overlapping between the following FFT input array in percentage; use a value between 5% and 95% (75% default)

tacq
Total acquisition time (in ms) to evaluate all the parameters for the time domain and frequency domain analysis in the same time.

tau
Time parameter to include for the moving root mean square (RMS) evaluation (for speed and/or acceleration); choose a value from: 25, 50, 100, 150, 250, 500 (default), 1000, 1500 and 2000.

subrng

Subrange FFT numbers to evaluate the frequency domain analysis results in each subrange frequency sector; choose a value from 8, 16 (default), 32, or 64 (this parameter is used by the condition monitoring project.

wind

Filter windowing type; choose from:

- 0 - Hanning (default)
- 1 - Hamming
- 2 - Flat Top

tdtype

Time domain datatype format:

- 0 - Speed RMS only
- 1 - Acceleration RMS only
- 2 - Speed RMS and Acceleration RMS

Once you have inserted the new parameters, the Command Line Interface prompts you to type [**y**] and press [**Enter**] to confirm the new values.

Figure 35. **Default parameter list and starting point**

```
 Stored STEVAL-IDP005V1 parameters are:
odr=6660 fs=4 hpf=3 size=2048 ovl=75 tacq=1000 tau=500 subrng=8 wind=0 tdtype=0

Enter new vibration parameters and/or press 'Enter'

New STEVAL-IDP005V1 parameters are:
odr=6660 fs=4 hpf=3 size=2048 ovl=75 tacq=1000 tau=500 subrng=8 wind=0 tdtype=0

Let's go ahead? ( [y/n] + 'Enter' )
```

The terminal emulator will show the following information:

- Time domain analysis X-Y-Z arrays according to the tdtype and tacq (timing window) parameters, transmitting the data every 5 ms. The figure below lists the following information:
  – the real ODR evaluated by the algorithm in order to have a more accurate value for the FFT arrays;
  – the time domain datalog with the timestamp in the first column, and the X-Y-Z value chosen, in order to plot the RMS speed trend mm/s.

Figure 36. **Time domain data**

```
ISM330DLC (Accelerometer): real ODR 6668.00 Hz


|     *** Time Domain Data ***      |
|   Time    |   Speed RMS [mm/s]    |
|   [ms]    |   X   |   Y   |   Z   |
|    129956 | 0.099 | 0.056 | 1.503 |
|    129961 | 0.096 | 0.054 | 1.503 |
|    129966 | 0.094 | 0.058 | 1.486 |
|    129971 | 0.092 | 0.061 | 1.484 |
|    129976 | 0.089 | 0.059 | 1.477 |
|    129981 | 0.089 | 0.061 | 1.473 |
|    129986 | 0.087 | 0.061 | 1.471 |
|    129991 | 0.088 | 0.059 | 1.468 |
|    129996 | 0.087 | 0.059 | 1.466 |
|    130001 | 0.088 | 0.059 | 1.464 |
|    130006 | 0.089 | 0.058 | 1.463 |
|    130011 | 0.089 | 0.059 | 1.463 |
|    130016 | 0.089 | 0.059 | 1.461 |
|    130021 | 0.088 | 0.059 | 1.460 |
```

- Frequency Domain X-Y-Z arrays according to the parameter settings for the configured timing window (tacq) as well as the bin frequency information. The output shows the acceleration power spectrum in m/s².

Figure 37. **Frequency domain data**

```
*** FFT Spectral Analysis ***
|binFreq|     Amplitude[m/s2]      |
|  [Hz] |   X    |   Y    |   Z    |
|   0.00|  0.000|  0.000|  0.000|
|   3.25|  0.000|  0.000|  0.000|
|   6.51|  0.000|  0.000|  0.000|
|   9.76|  0.000|  0.000|  0.000|
|  13.02|  0.001|  0.000|  0.001|
|  16.27|  0.001|  0.001|  0.001|
|  19.53|  0.001|  0.001|  0.001|
|  22.78|  0.001|  0.001|  0.001|
|  26.04|  0.001|  0.001|  0.002|
|  29.29|  0.002|  0.003|  0.002|
|  32.54|  0.008|  0.015|  0.003|
|  35.80|  0.009|  0.016|  0.003|
|  39.05|  0.003|  0.004|  0.002|
|  42.31|  0.002|  0.003|  0.002|
|  45.56|  0.001|  0.002|  0.005|
|  48.82|  0.002|  0.002|  0.019|
|  52.07|  0.002|  0.001|  0.015|
```

- Frequency Domain final results, including the average number of FFTs used during processing processing.

Figure 38. **FFT results**

```
*** FFT Results ***

 FFT AVERAGE Number = 11

Max_Amplitudes  -> Xa:   1.701| Ya:   0.160| Za: 14.551|
Bin_Frequencies -> Xf:575.855| Yf:575.855| Zf:575.855|
```

- The maximum X-Y-Z acceleration peak.

Figure 39. **Maximum X-Y-Z acceleration peak**

```
*** Acceleration Time Domain Max Peak ***
| Xpk:   0.168| Ypk:   0.125| Zpk:   0.122|
```

- The final step lets you change some parameters again and run a new analysis.

Figure 40. **Summary window**

```
|##################### Next Measurement #####################

 Stored STEVAL-IDP005V1 parameters are:
odr=6660 fs=4 hpf=3 size=2048 ovl=75 tacq=1000 tau=500 subrng=8 wind=0 tdtype=0

Enter new vibration parameters and/or press 'Enter'
```

9.4 Condition Monitoring via Service UART

The terminal emulator will show the following information:

Figure 41. **Condition Monitoring header log**

```
STEVAL-IDP005V1 [Application - Condition Monitoring - FW v.1.0.0]
2018 STMicroelectronics

MCU ID....: 0x393832383035511900400038
MCU SYSCLK: 180000000 Hz
MCU HCLK..: 180000000 Hz
MCU PCLK1.: 45000000 Hz
MCU PCLK2.: 90000000 Hz

Stored STEVAL-IDP005V1 parameters are:
odr=6660 fs=4 hpf=3 size=2048 ovl=75 tacq=5000 tau=500 subrng=32 wind=0 tdtype=0


Enter new vibration parameters and/or press 'Enter'
```

This project includes the components developed for vibration analysis, environmental measurement and a specific frequency domain analysis that uses subranges to show the harmonics contributing across the power spectrum bandwidth.

- The environmental sensor measurements are listed below:

Figure 42. **Environmental data**

```
*** ENVIRONMENTAL DATA VALUES ***

LPS22HB_T: 27.80   Deg
LPS22HB_P: 1011.65 mBar
HTS221_T : 28.10   Deg
HTS221_H : 39.40   rH
```

- Frequency domain results are grouped into subranges according to the subrng parameter, which is more useful for vibration analysis that can also verify the relative maximum values across the frequency bandwidth. The information is provided for each axis, with the frequency and maximum amplitude for each subrange.

Figure 43. **Frequency domain analysis with subranges**

```
*** FFT SUBRANGE RESULTS ON THREE AXES X-Y-Z ***

| 101.144|  0.016 |  35.890|  0.019 |  48.940|  0.019|
| 499.192|  0.305 | 499.192|  0.054 | 499.192|  6.529|
| 913.555|  0.003 | 880.928|  0.003 | 998.385|  0.005|
|1252.875|  0.003 |1435.586|  0.002 |1497.577|  0.007|
|2035.922|  0.002 |1716.178|  0.002 |1876.050|  0.002|
|2140.328|  0.003 |2101.176|  0.002 |2215.370|  0.002|
|2871.172|  0.002 |2567.741|  0.001 |2841.808|  0.003|
|2975.578|  0.003 |2942.951|  0.001 |2923.375|  0.001|
```

- The final step lets you change some parameters again and run a new analysis.

9.5 Predictive Maintenance via Service UART

Predictive Maintenance is based on continuous comparison of vibration data with threshold values, which may be provided by the machinery manufacturer. The objective is to monitor potentially damaging conditions that cannot be identified in conventional scheduled maintenance.

The STEVAL-IDP005V1 firmware lets you modify time domain and frequency domain conditions:

1. Time domain thresholds with three different warning thresholds and three different alarm thresholds to continuously compare against the following processed data:

   – Speed RMS
   – Acceleration Peak
2. Frequency domain thresholds with warning and alarm thresholds for all the subranges. The thresholds can be set using the command line interface, while the threshold values are stored in the MotionSP_thresholds.h file.

When you run the application, the terminal window will show the following results:

- Time Domain thresholds status for the X-Y-Z RMS speed Status, with values derived from the comparisons.

Figure 44. RMS speed threshold status

```
*** TIME DOMAIN SPEED RMS THRESHOLDS STATUS X-Y-Z ***

|   GOOD    |   GOOD    |   GOOD    |
|   0.093   |   0.059   |   1.343   |
```

- Time Domain threshold status for the X-Y-Z acceleration peak, with values derived from the comparisons.

Figure 45. Acceleration peak threshold status

```
*** TIME DOMAIN ACC PEAK THRESHOLDS STATUS X-Y-Z ***

|   GOOD    |   GOOD    |  WARNING |
|   0.488   |   0.314   |   6.254  |
```

- Frequency domain warning and alarm thresholds status for all axes and for each configured subrange, including the relative maximum value detected in terms of frequency and amplitude. The following figure shows an example with subrng=8 and an alarm condition in the second subrange on the z axis.

Figure 46. Frequency domain subranges threshold status

```
*** FREQUENCY DOMAIN THRESHOLDS STATUS ON SUBRANGE vs X-Y-Z ***

|          GOOD        |          GOOD        |          GOOD        |
| 101.07  |      0.020 |  32.60 |      0.014 |  48.90 |      0.019 |

|          GOOD        |          GOOD        |         WARNING      |
| 498.82  |      0.371 | 498.82 |      0.170 | 498.82 |      5.885 |

|          GOOD        |          GOOD        |          GOOD        |
|1000.90  |      0.007 |1176.95 |      0.002 |1000.90 |      0.003 |

|          GOOD        |          GOOD        |          GOOD        |
|1499.72  |      0.009 |1499.72 |      0.004 |1499.72 |      0.006 |

|          GOOD        |          GOOD        |          GOOD        |
|2080.04  |      0.002 |1780.10 |      0.002 |1731.19 |      0.002 |

|          GOOD        |          GOOD        |          GOOD        |
|2190.89  |      0.002 |2112.64 |      0.002 |2393.03 |      0.002 |

|          GOOD        |          GOOD        |          GOOD        |
|2869.02  |      0.003 |2624.50 |      0.002 |2839.68 |      0.002 |

|          GOOD        |          GOOD        |          GOOD        |
|2921.19  |      0.002 |2960.31 |      0.001 |2960.31 |      0.001 |
```

- Next, the output shows general status messages related to time or frequency domain comparisons with thresholds, as shown below:

Figure 47. Threshold status summary

```
|################## TIME DOMAIN STATUS ####################
|     -----------> WARNING<-----------      |
|################## FREQUENCY DOMAIN STATUS ###############
|     -----------> WARNING<-----------      |
```

- The final step lets you change some parameters again and run a new analysis.

10 How to run projects via IO-Link

The STEVAL-IDP005V1 is also able to communicate through its embedded IO-Link PHY device, so the board can receive and transmit data and commands to and from the STEVAL-IDP004V1 master board based on the IO-Link PHY master.

In the firmware package, the CondMonitor_IOL project (in STSW-BFA001V1\Projects\Demonstrations\Condition_Monitoring\CondMonitor_IOL) can communicate via IO-Link using dedicated functions (no stack libraries are implemented) to package the post processing results and sensor parameters. With IO-Link connectivity, the project can also output results to a GUI.

Follow the procedure below to run the application with IO-Link:

Step 1. Connect the STEVAL-IDP005V1 to the STEVAL-IDP004V1 IO-Link master board using a standard 4-wire cable with M12 A-coded 4-pin connectors, male on one end and female on the other.

Step 2. Connect the STEVAL-IDP004V1 to the power supply @ Vin = 18 to 32 V.

Step 3. Connect the STEVAL-IDP005V1 to the STEVAL-UKI001V1 and update the firmware.
Use the binary file in STSW-BFA001V1\Projects\Demonstrations\Condition_Monitoring\CondMonitor_IOL\Binary

Step 4. Turn on the power supply for the IO-Link master board and update the STEVAL-IDP005V1 firmware.
Use the STEVAL_IDP005V1_CondMonitor_IOL.bin or *.hex binary file

Step 5. Disconnect the assembly used for the firmware update, but leave the two boards with IO-Link and connect the RS-485 adapter for USB.

Step 6. Connect the USB cable to your PC and run the GUI to experience the functionality as condition monitoring by Service UART.

RELATED LINKS

5.1 Connection through an ST-LINK/V2-1 on page 18
5.2 Connection through an STEVAL-IDP004V1 on page 19
11 Graphical Interface overview on page 39

11 Graphical Interface overview

The tool is designed to let you simultaneously monitor the different values measured by each sensor node connected to the STEVAL-IDP004V1 IO-Link master board.

The GUI handles commands and data exchange in string format between a PC and the STEVAL-IDP004V1. Each command received by the master is processed into byte format and sent to the sensor node.

As the sensor node has several sensors, a set of commands are available to show information like humidity and pressure values, vibration frequency spectra and time domain acceleration analyses.

In the [**Vibration Analysis**] tab, you can select one of the following analyses:

- ENV for environmental data
- RMS/PEAK for time domain results
- ACC FFT for frequency spectrum results

RELATED LINKS

8.4.3 Demonstrations folder on page 27

10 How to run projects via IO-Link on page 38

5.2 Connection through an STEVAL-IDP004V1 on page 19

11.1 Data commands for sensor queries

Data communication between the STEVAL-IDP005V1 and STEVAL-IDP004V1 is managed through a simple serial connection at 230.4 kbaud.

Communication is initiated by the master node with a data frame signaling the STEVAL-IDP005V1 that a communication request has been received by the host. The sensor node flags the request and sends the appropriate data when it has been processed and ready to be sent.

The communication commands are defined in the Master_DeviceCOMM.h file in the STEVAL-IDP004V1 firmware. The command structures are shown below.

1. FRAME_TYPE_CMD (0x21): this command is sent from master to the device. It communicates that a command will be sent to the sensor node, which will return an acknowledge byte ACK_CMD when the sensor node is ready. Once communication initiates, the master node can send the following requests to the sensor:
   – GET_SENSOR_TYPE (0x38): requests the sensor type and FW version on the board.
   – GET_ACC_ RAW (0x31): requests accelerometer data from the sensor.
   – GET_ACC_TDM (0x32): requests time domain data from the sensor (Peak,RMS).
   – GET_ENV_MEASURE (0x33): requests accelerometer data from the sensor.
   – GET_ACC_FFT (0x36): requests vibration power spectrum.
   – GET_SENSOR_MCU_ID (0x3C): requests the MCU ID of the sensor.
2. FRAME_TYPE_DATA (0x22): this command is sent from master to the device. It communicates that a data frame will be sent to the sensor node, which will return an acknowledge byte ACK_DATA when the sensor node when is ready. Once communication initiates, the master send the following commands to the sensor:
   – SET_PRM_CPT (0x40): sets the computation parameter for time domain calculation.
   – SET_PRM_ACC (0x41): sets the accelerometer data acquisition parameters (ODR, operating mode and filtering frequency).

Both sets of data are stored in the flash. The microcontroller reads the locations and updates its own settings after a reset.

11.3 How to use the STEVAL-IDP005V1 GUI interface

To perform this task, your PC must be connected with the Demonstration kit via the RS485 cable.

Follow the steps below to exchange data with the sensor node:

Step 1. Set PC-Master Board communication parameters.
- Name: COM Port name
- Baud Rate: 230400 Baud
- Data:8
- Parity: None
- Stop Bit: One
- Flow Control: None

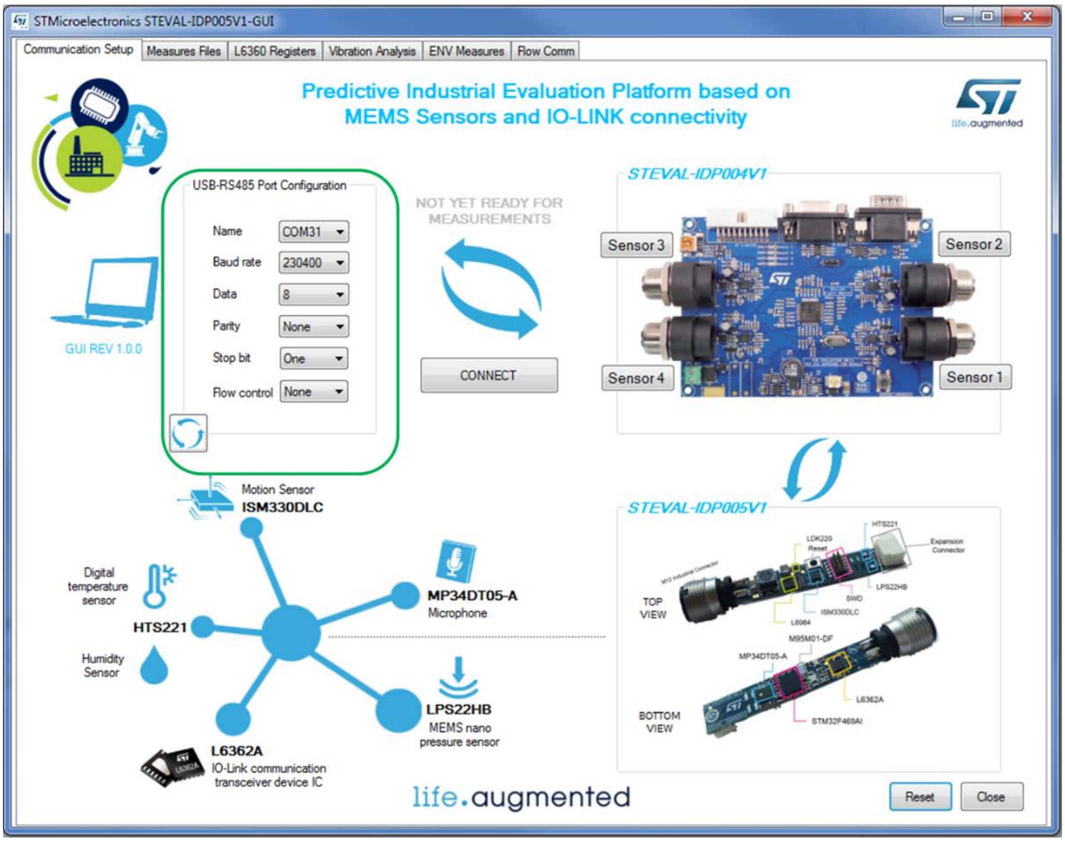
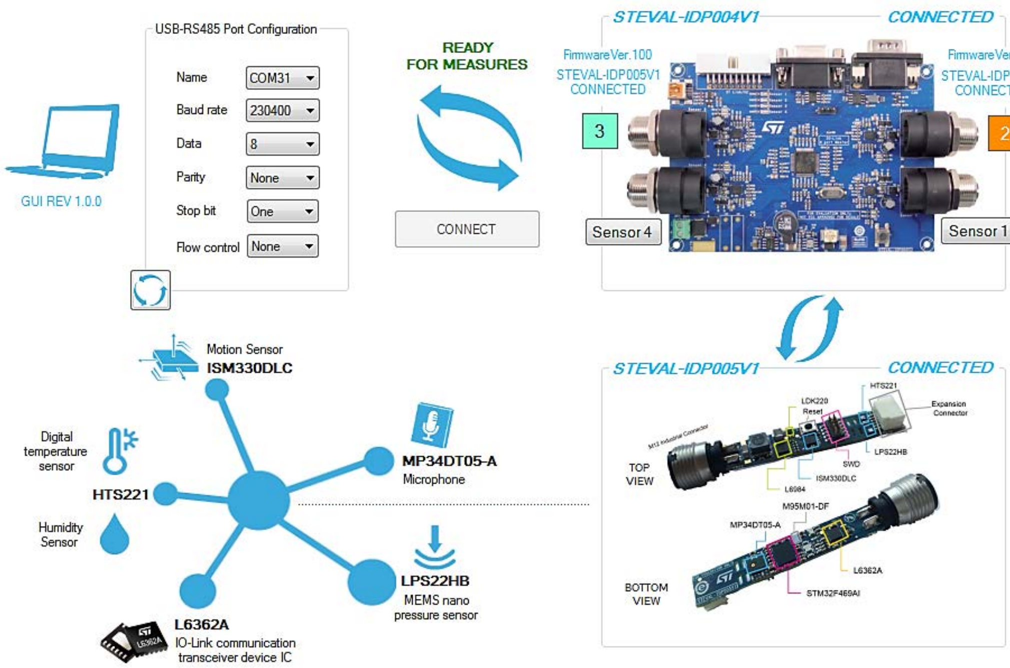


Figure 48. **Communication parameter settings**

Step 2. Click on one or more [**Sensor**] tabs according to the connected devices in the master board section (e.g., [**Sensor 2**] and [**Sensor 3**]) and click [**Connect**].

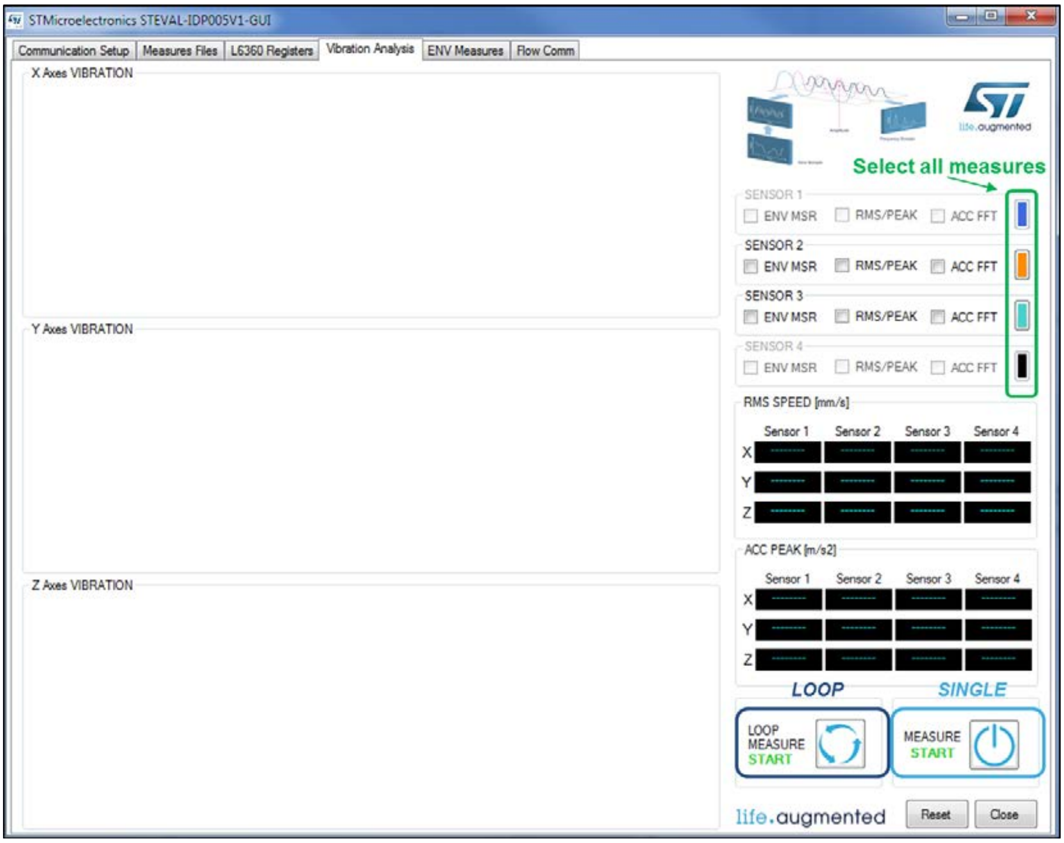
Figure 49. **STEVAL-IDP005V1 home page connection**



In this phase, the GUI programs the master IC with the default configuration and then sends the command IDS to identify the sensor node for each port on the network. If the sensor is recognized, the corresponding button on the GUI changes color and the “CONNECTED” label appears with firmware version information.

Step 3. Select the [**Vibration Analysis**] tab.

Figure 50. Vibration analysis tab



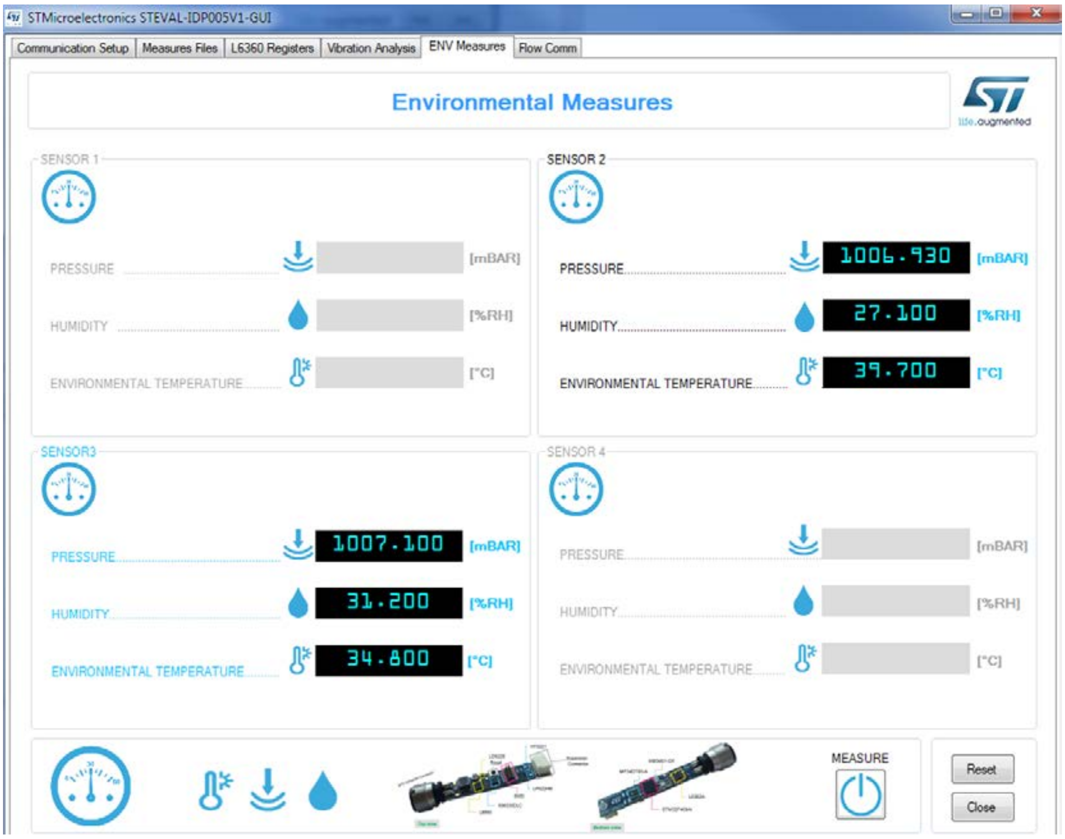
Step 4. Check one or more of the following fields for each active sensor that you wish to analyze:
- ENV MSR (for environmental data shown in the [**ENV Measures**] tab)
- RMS/PEAK (for time domain analysis values in 3D dedicated sector)
- ACC FFT (for frequency domain analysis, available in 3D plot)

You can select all of the analyses by clicking on the button to the right of the fields.

Step 5. Select one of the following options to run the application:
- [**MEASURE START**] for a single pass
- [**LOOP MEASURE START**] for loop mode acquisition

Figure 51. Frequency domain and time domain results

Figure 52. Environmental Measurement results



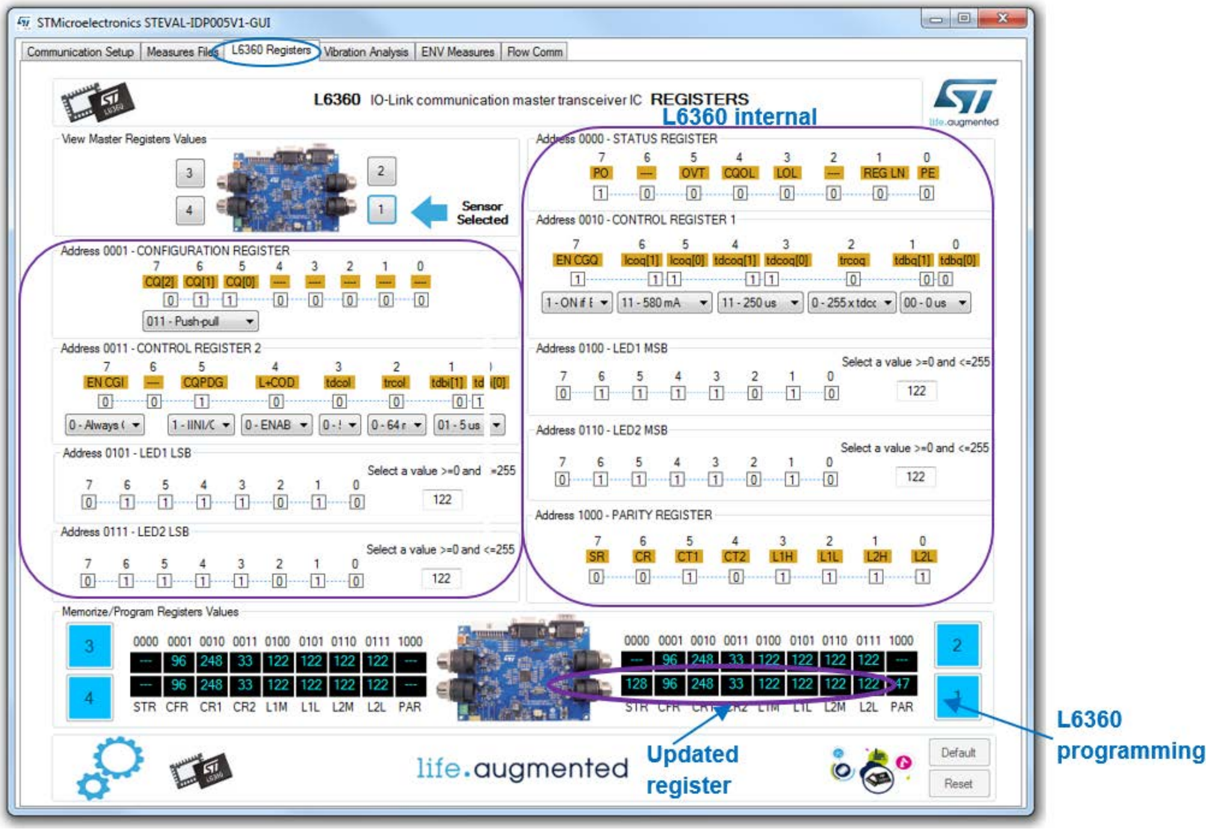
11.3.1 How to modify the default L6360 settings

Step 1. Press [**Connect**] to connect the PC and the master board.

Step 2. Select the [**L6360 Registers**] tab.

Step 3. Click on the relevant Master port in the [**View Master Registers Values**] section
This will call up the current IC register settings for the selected port.

Figure 53. **L6360 register update**



Step 4. Change the register settings.

Once the configuration has been changed, the decimal format is updated in the [**Memorize/Program Registers Values**] section.

Step 5. Click on the blue button for the modified Master in the [**Memorize/Program Registers Values**] section.

11.3.2 How to save the log files

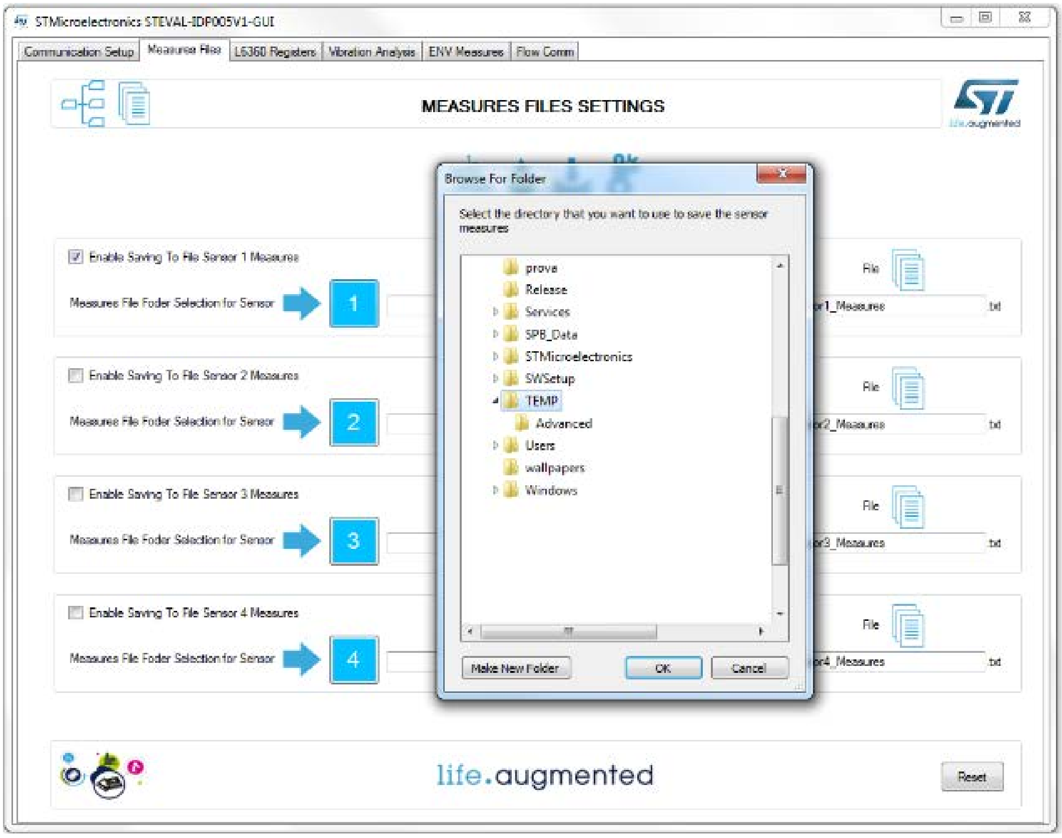
The [**Flow**] [**Comm**] tab shows the command and data communication history during the session with the GUI. Follow the procedure below to store the communication history in log file

Step 1. Select the [**Measures Files**] tab.

Step 2. Check the [**Enable Saving**] [**To**] [**File Sensor X Measures**] box.

Step 3. Click the corresponding square blue button and select the folder path and file name for the log file.

Figure 54. Log file storage



Step 4. Run the analysis in single or loop mode.

In loop mode, each measurement does not overwrite the previous run.

When the sequence is completed, the log for is saved to the file.

12 Bill of materials

Table 3. **Bill of materials**

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part Number
1	1	C1	100nF, 50V, ±10%	X7R, SMD 0402	TDK	CGA2B3X7R1H104K050BB
2	1	C2	3.3uF, 50V, ±10%	X7R, SMD 1206	TDK	C3216X7R1H335K160AC
3	5	C3, C5, C6, C19, C31	1uF, 10V, ±10%	X7S, SMD 0402	TDK	C1005X7S1A105K050BC
4	1	C4	22uF, 16V, ±20%	X7R, SMD 1210	MULTICOMP	MC1210B226M160CT
5	1	C7	47nF, 10V, ±10%	X7R, SMD 0402	Murata	GRM155R71A473KA01D
6	1	C8	1uF, 50V, ±10%	X5R, SMD 0603	Murata	GRT188R61H105KE13D
7	9	C9, C10, C11, C12, C23, C26, C44, C45, C48	10pF, 10V, ±1%	C0G, SMD 0402	MULTICOMP	MCMT15N100F100CT
8	1	C13	TBD, 10V,	SMD 0402		
9	17	C14, C17, C18, C20, C22, C28, C32, C34, C35, C36, C37, C38, C39, C40, C41, C46, C47	100nF, 10V, ±10%	X7R, SMD 0402	Wurth Elektronik	885012205018
10	1	C15	100nF, 10V, ±10%	X7R, SMD 0805	TDK	C1005X7R1A104K050BB
11	2	C21, C33	4.7uF, 10V, ±10%	X7S, SMD 0603	TDK	C1608X7S1A475K080AC
12	2	C24, C25	10pF, 50V, ±5%	C0G, SMD 0402	Kemet	C0402C100J5GACTU
13	3	C27, C29, C30	2.2uF, 10V, ±10%	X7R, SMD 0402	Murata	GRM155Z71A225KE44D
14	2	C42, C43	16pF, 50V, ±5%	C0G, SMD 0402	Murata	GRM1555C1H160JA01D
15	2	D1, D2	NSR05T40P2	Schottky Barrier Diode, SOD-923	On Semiconductor	NSR05T40P2T5G
16	1	JP1	IO-Link CONN	IO-Link 4 position M12 A-coded connector	Binder	9043121204
17	1	J1	SWD Connector	SMT Pitch 1.27 mm (5x2)	Samtec	FTS-105-01-L-DV
18	1	J2	Auxiliary Connector	SMT Pitch 1 mm (8x6.8)	JST Sales America Inc.	SM06B-NSHSS-TB
19	1	L1	68uH, Isat = 0.4 A / Rdc = 0.34 ohm, ±30%	Shielded Power Inductor, SMD (4.8x4.8x2.8 mm)	Wurth Elektronik	744043680
20	1	R1	1M, 0.1 W, ±1%	SMD 0402	Any	

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part Number
21	3	R2, R22, R23	100k, 0.1 W, ±1%	SMD 0402	Any	
22	1	R3	909k, 0.1 W, ±1%	SMD 0402	Any	
23	1	R4	205k, 0.1 W, ±1%	SMD 0402	Any	
24	1	R5	68k, 0.1 W, ±1%	SMD 0402	Any	
25	1	R6	100k, 0.1 W, ±1%	SMD 0402	Any	
26	2	R7, R8	4k7, 0.1 W, ±1%	SMD 0402	Any	
27	5	R9, R10, R11, R13, R25	100R, 0.1 W, ±1%	SMD 0402	Any	
28	1	R12	22k, 0.1 W, ±1%	SMD 0603	Any	
29	5	R16, R17, R24, SB9, SB10	10k, 0.1 W, ±1%	SMD 0402	Any	
30	5	R18, R19, R20, R21, R26	2k, 0.1 W, ±1%	SMD 0402	Any	
31	2	SB2, SB5	0R, 0.1 W, ±1%	SMD 0402	Any	
32	3	SB6, SB7, SB8	0R, 0.1 W, ±1%	SMD 0402	Any	
33	1	SW1	Reset	smd (L 4.6 x W 2.2 x H 1.9 mm)	C & K	KMR211G LFS
34	1	U1	L6984	Step-Down Switching Regulator, VDFPN10 (3x3x1.0 mm)	ST	L6984ATR
35	1	U2	LDK220	LDO, DFN6 (1.2x1.3x0.5 mm)	ST	LDK220PU33R
36	1	U3	L6362A	IO-Link Communication Transceiver, VFDFPN 12L (3x3x0.90 mm)	ST	L6362ATR
37	1	U4	ISM330DLC	3D Accelerometer, LGA-14L (2.5x3x0.83 mm)	ST	ISM330DLCTR
38	1	U6	HTS221	Humidity and Temperature Sensor, HLGA-6L (2x2x0.9 mm)	ST	HTS221TR
39	1	U7	MP34DT05-A	Microphone, HCLGA-4LD (3x4x1 mm)	ST	MP34DT05TR-A
40	1	U8	LPS22HB	Pressure Sensor, HLGA-10L (2x2x0.76 mm)	ST	LPS22HBTR

Item	Q.ty	Ref.	Value	Description	Manufacturer	Part Number
41	1	U9	M95M01-DF	EEPROM, WLCSP8 (2.578x1.716 mm)	ST	M95M01-DFCS6TP/K
42	1	U10	STM32F469AI	ARM®Cortex®-M4 32-bit MCU, WLCSP 168L DIE 434 12X14 P 0.4mm	ST	STM32F469AIY6TR
43	1	Y1	32.768 kHz, ±20ppm	Crystal, smd (2.05x1.2x0.55mm)	NDK	NX2012SA 32.768kHz EXS00A-MU00389
44	1	Y2	24 MHz, ±20ppm	Crystal, smd (2x1.6x0.45 mm)	NDK	NX2016SA 24.000MHz EXS00A-CS05544

13 Schematic diagrams

Figure 55. STEVAL-IDP005V1 schematic – General purpose industrial sensor

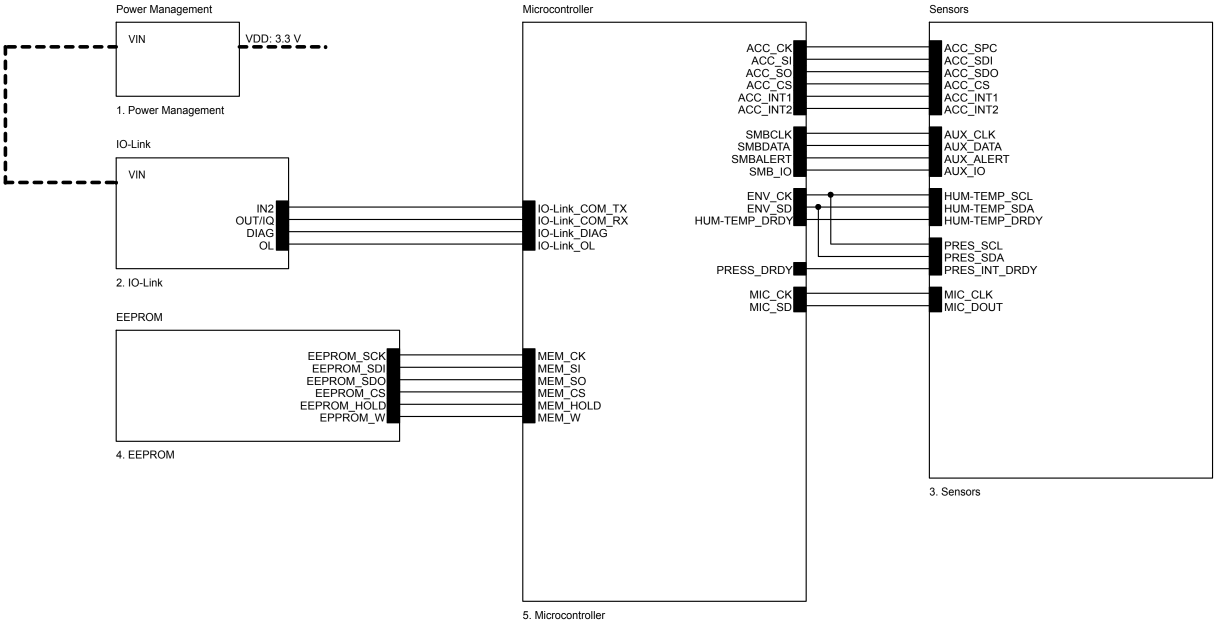


Figure 56. STEVAL-IDP005V1 schematic – Power management

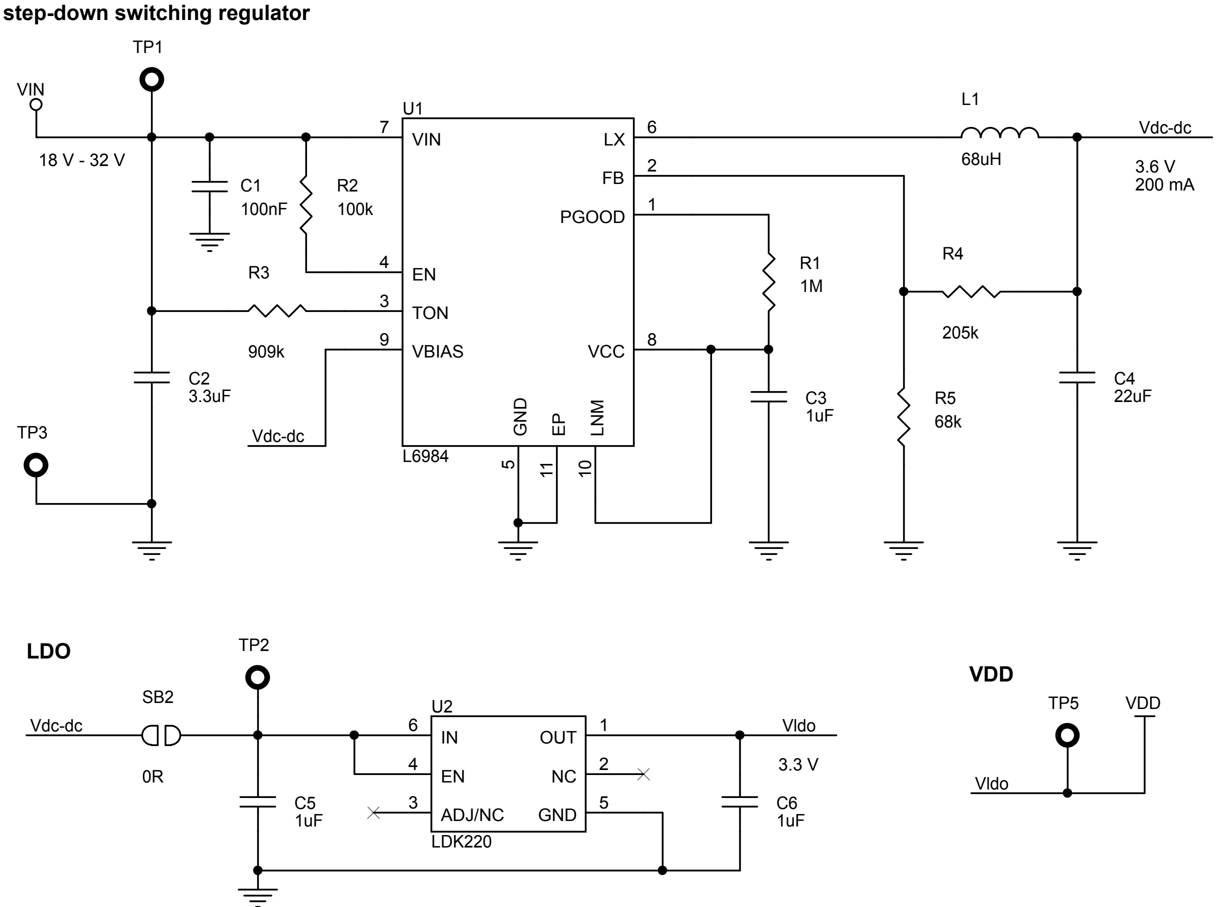


Figure 57. STEVAL-IDP005V1 schematic – IO-Link

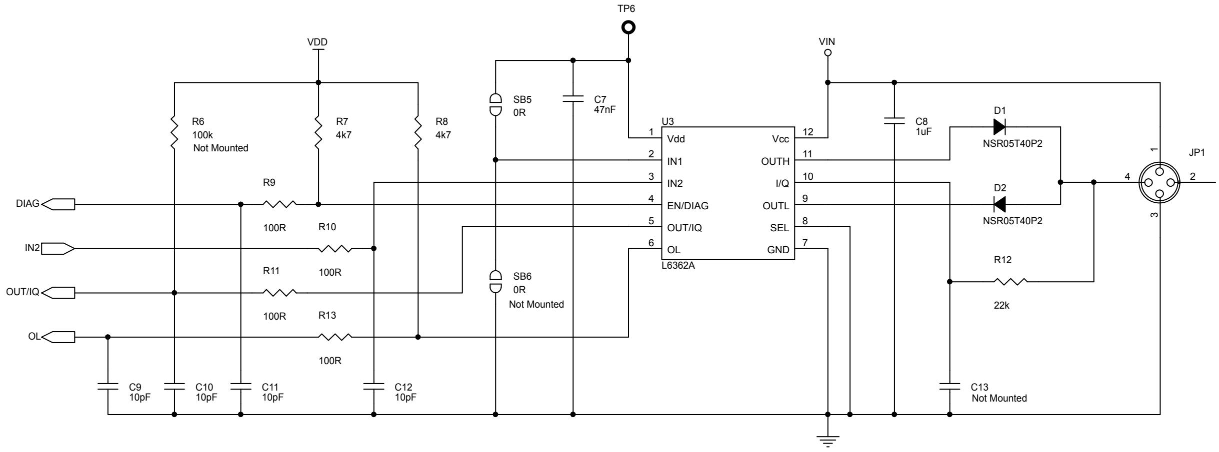


Figure 58. STEVAL-IDP005V1 schematic – Sensors

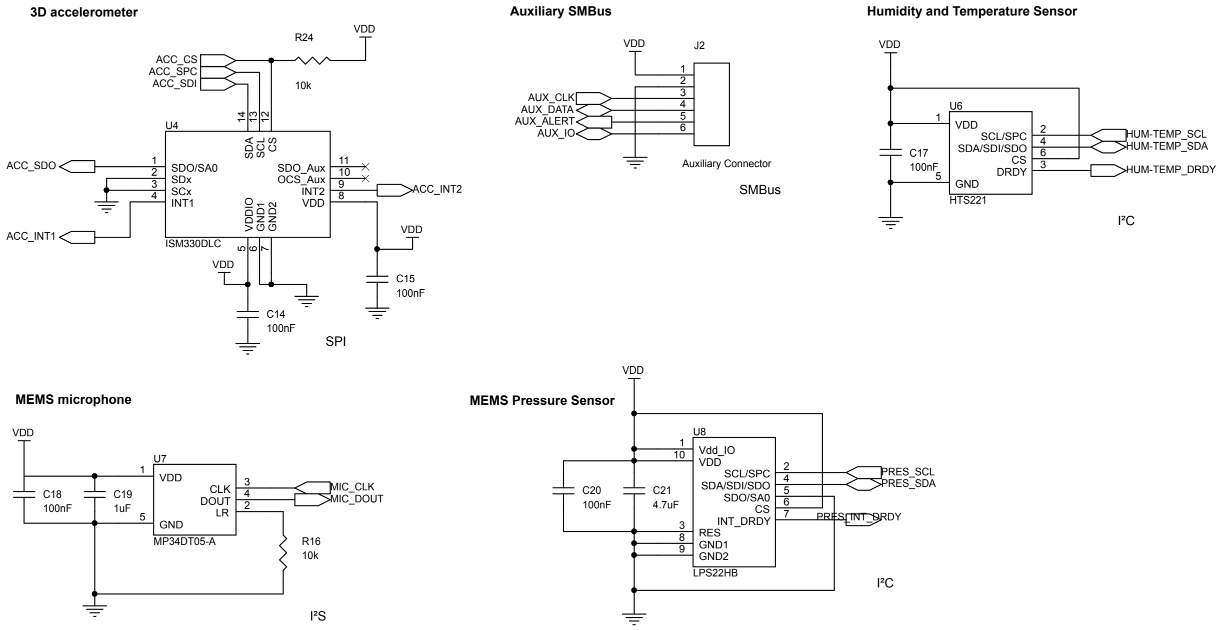


Figure 59. STEVAL-IDP005V1 schematic – EEPROM

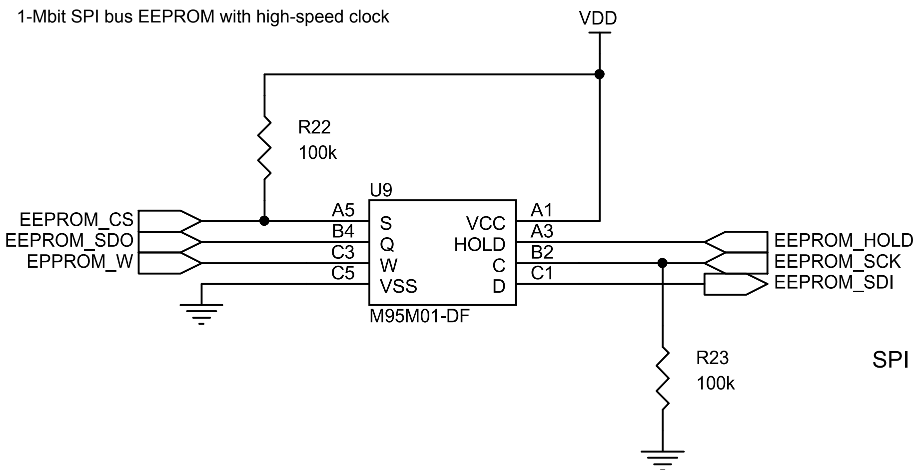


Figure 60. STEVAL-IDP005V1 schematic – Microcontroller (part A)

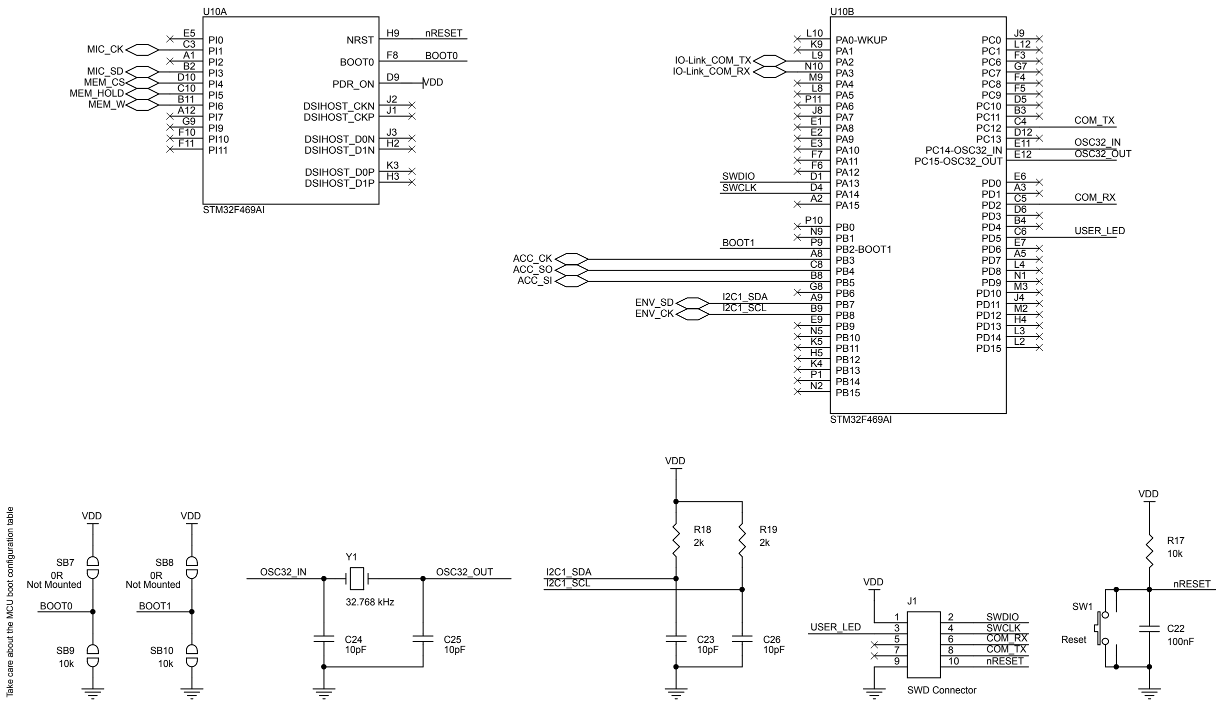
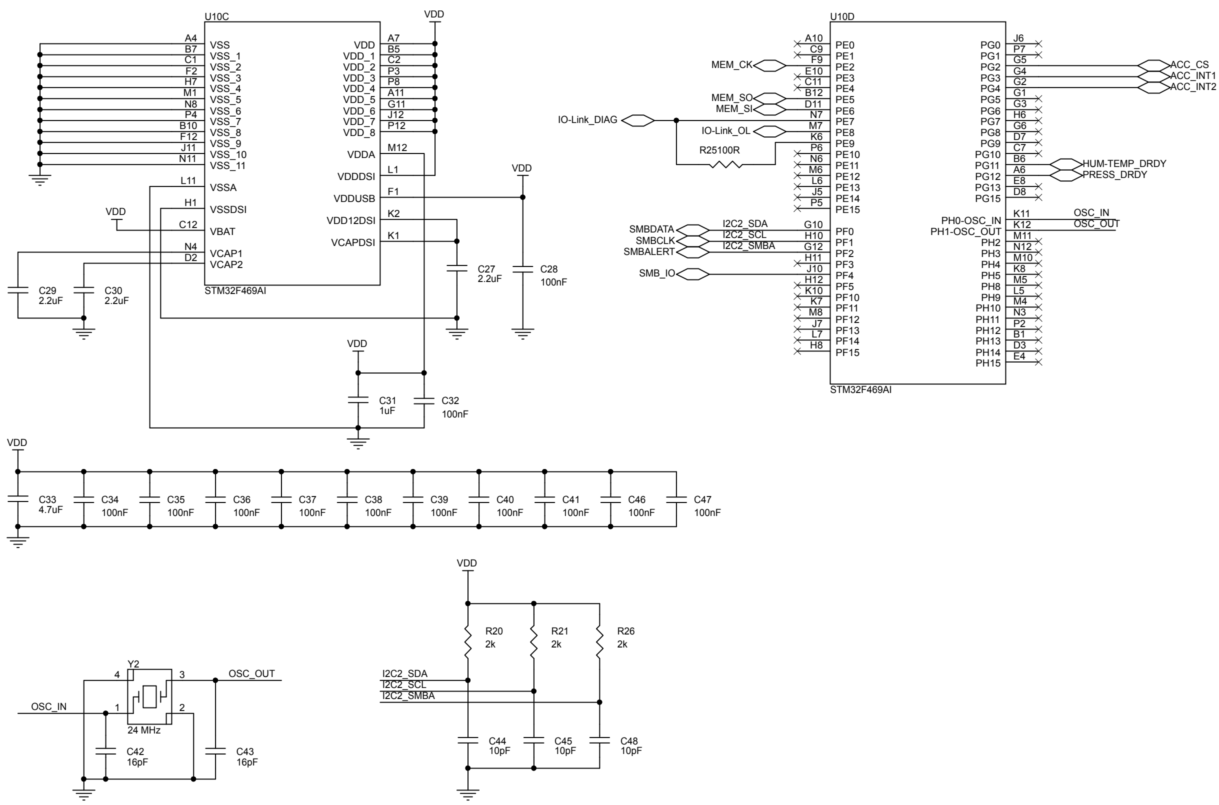


Figure 61. STEVAL-IDP005V1 schematic – Microcontroller (part B)



Revision history

Table 4. **Document revision history**

Date	Version	Changes
19-Jul-2018	1	Initial release.

Contents

List of figures

List of tables