
How to use a sensor on a DIL 24 socket in X-CUBE-MEMS1 package applications

Introduction

The **X-CUBE-MEMS1** software package allows you to build applications based on sensor data from STM32 Nucleo development platforms mounting a compatible X-NUCLEO expansion board with several high performance ST motion MEMS and environmental sensors.

You can, however, substitute individual sensors on the X-NUCLEO expansion board with single sensor boards with compatible pins for the standard DIL 24 socket.

1 Step by step procedure to replace an on-board sensor with a DIL 24 sensor

In this example, we replace the **LSM6DSL** sensor on the **X-NUCLEO-IKS01A2** expansion board with the **LSM6DSO** sensor connected to a board with a DIL 24 socket.

We have chosen the DataLogFusion application (using MotionFX sensor fusion algorithm) running on the **NUCLEO-L476RG** development board.

With minor modifications, you can follow the same procedure for other sensors, expansion boards and **STM32 Nucleo** board platforms as well.

Step 1. Add the sensor in the board configuration header file.

In the `inc/iks01a2_conf.h` file, disable the use of the **LSM6DSL** sensor by inserting "0U" as highlighted below in bold, and enable the **LSM6DSO** sensor by adding the define statement also shown below in bold.

```
#define USE_IKS01A2_ENV_SENSOR_HTS221_0      1U
#define USE_IKS01A2_ENV_SENSOR_LPS22HB_0    1U

#define USE_IKS01A2_MOTION_SENSOR_LSM6DSL_0  0U
#define USE_IKS01A2_MOTION_SENSOR_LSM303AGR_ACC_0  1U
#define USE_IKS01A2_MOTION_SENSOR_LSM303AGR_MAG_0  1U

#define USE_IKS01A2_MOTION_SENSOR_LSM6DSO_0      1U
```

Step 2. Add the sensor drivers in a project for a specific IDE (for example, SW4STM32) and include the path to the sensor drivers into the project settings.

Figure 1. Add LSM6DSO sensor drivers in a project for the SW4STM32 IDE

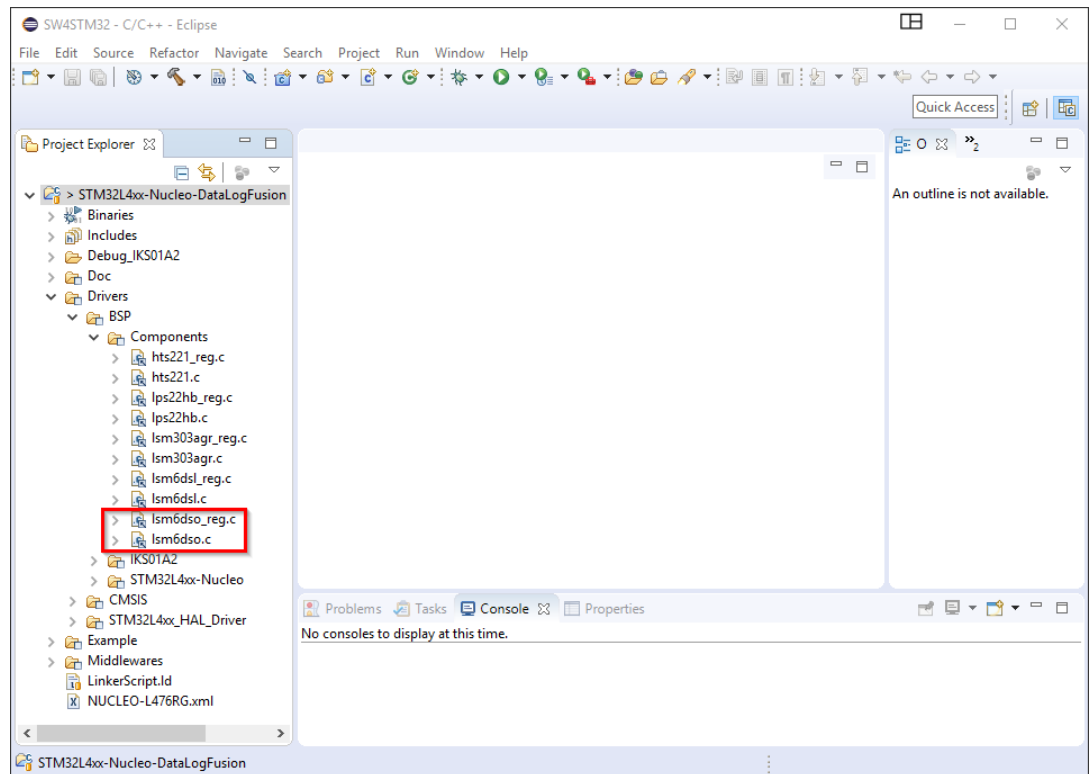
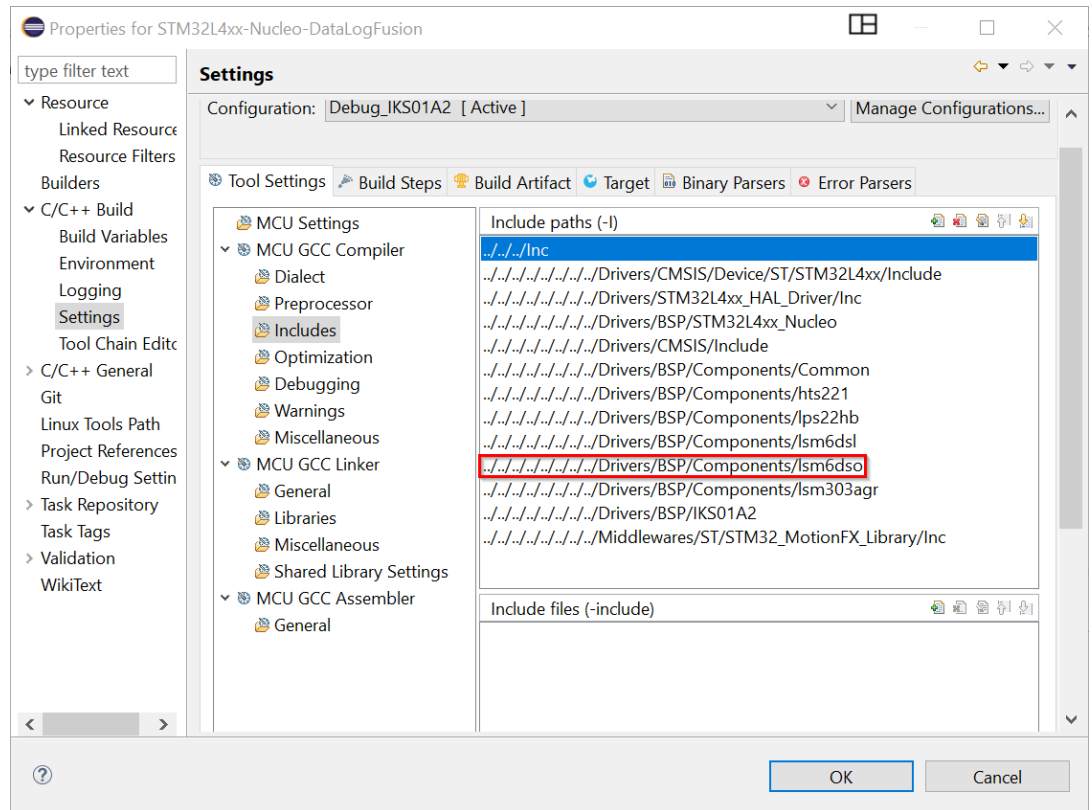


Figure 2. Add LSM6DSO sensor including path to a project for the SW4STM32 IDE


Step 3. Substitute the sensor on the X-NUCLEO expansion board with the single sensor on the DIL 24 socket board in the **X-CUBE-MEMS1** application.

Step 3a. Substitute the **LSM6DSL** sensor handler with the **LSM6DSO** sensor handler in the *src/main.c* file, *main()* function (in bold below).

```

/* Acquire data from enabled sensors and fill Msg stream */
RTC_Handler(&msg_dat);
Accelero_Sensor_Handler(&msg_dat, IKS01A2_LSM6DSO_0);
Gyro_Sensor_Handler(&msg_dat, IKS01A2_LSM6DSO_0);
Magneto_Sensor_Handler(&msg_dat, IKS01A2_LSM303AGR_MAG_0);
Humidity_Sensor_Handler(&msg_dat, IKS01A2_HTS221_0);
Temperature_Sensor_Handler(&msg_dat, IKS01A2_HTS221_0);
Pressure_Sensor_Handler(&msg_dat, IKS01A2_LPS22HB_0);
    
```

Step 3b. Substitute the **LSM6DSL** sensor initialization with **LSM6DSO** initialization in the *src/main.c* file, *Init_Sensors()* function (in bold below).

```

/**
 * @brief Initialize all sensors
 * @param None
 * @retval None
 */
static void Init_Sensors(void)
{
    (void)IKS01A2_MOTION_SENSOR_Init(IKS01A2_LSM6DSO_0, MOTION_ACCELERO | MOTION_GYRO);
    (void)IKS01A2_MOTION_SENSOR_Init(IKS01A2_LSM303AGR_MAG_0, MOTION_MAGNETO);
;
    (void)IKS01A2_ENV_SENSOR_Init(IKS01A2_HTS221_0, ENV_TEMPERATURE | ENV_HUMIDITY);
    (void)IKS01A2_ENV_SENSOR_Init(IKS01A2_LPS22HB_0, ENV_PRESSURE);
}
    
```

Step 3c. Add the **LSM6DSO** sensor Unicleo ID from the table below to the *src/DemoSerial.c* file (in bold below the table).

Table 1. Unicleo-GUI supported sensors and IDs

Accelerometer/gyroscope		Magnetometer		Pressure/temperature		Humidity/temperature	
Name	ID	Name	ID	Name	ID	Name	ID
LSM6DSO	1	LIS3MDL	1	LPS25HB	1	HTS221	1
LSM6DS3	2	LSM303AGR_MAG	2	LPS22HB	3	STTS751	2
LSM6DSL	3	LIS2MDL	3	LPS22HH	4		
LIS2DH12	4						
H3LIS331DL	5						
LSM303AGR	6						
LSM6DSO	7						
LIS2DW12	8						
LSM6DSR	9						
LSM6DSOX	10						

```
#define LPS25HB_UNICLEO_ID_ONBOARD 1
#define LPS25HB_UNICLEO_ID_DIL 2
#define LPS22HB_UNICLEO_ID 3
#define HTS221_UNICLEO_ID 1
#define LSM6DS0_UNICLEO_ID 1
#define LSM6DS3_UNICLEO_ID 2
#define LSM6DSL_UNICLEO_ID 3
#define LSM6DSO_UNICLEO_ID 7
#define LIS3MDL_UNICLEO_ID 1
#define LSM303AGR_UNICLEO_ID_MAG 2
```

Step 3d. Substitute **LSM6DSL** sensor disable with **LSM6DSO** disable in the *src/DemoSerial.c* file, `HandleMSG()` function, `CMD_ACCELERO_GYRO_Init` case (in bold below).

```
Serialize_s32(&Msg->Data[3], LSM6DSO_UNICLEO_ID, 4);
```

Step 3e. Substitute **LSM6DSL** sensor enable with **LSM6DSO** enable in the *src/DemoSerial.c* file, `HandleMSG()` function, `CMD_Start_Data_Streaming` case (in bold below).

```
if ((SensorsEnabled & ACCELEROMETER_SENSOR) == ACCELEROMETER_SENSOR)
{
    (void)IKS01A2_MOTION_SENSOR_Enable(IKS01A2_LSM6DSO_0, MOTION_ACCELERO);
}
if ((SensorsEnabled & GYROSCOPE_SENSOR) == GYROSCOPE_SENSOR)
{
    (void)IKS01A2_MOTION_SENSOR_Enable(IKS01A2_LSM6DSO_0, MOTION_GYRO);
}
```

Step 3f. Substitute **LSM6DSL** sensor disable with **LSM6DSO** disable in the *src/DemoSerial.c* file, `HandleMSG()` function, `CMD_Stop_Data_Streaming` case (in bold below).

```
(void)IKS01A2_MOTION_SENSOR_Disable(IKS01A2_LSM6DSO_0, MOTION_ACCELERO);
(void)IKS01A2_MOTION_SENSOR_Disable(IKS01A2_LSM6DSO_0, MOTION_GYRO);
```

Step 4. Change sensor orientation (optional).

In the *MotionFX_Manager.c* file `MotionFX_manager_init()` function, check accelerometer, gyroscope and magnetometer sensor orientation setup with real DIL 24 sensor orientation and update if necessary.

In our case, we need to check accelerometer and gyroscope sensor orientation setup with **LSM6DSO** (DIL 24 module) orientation (in bold below).

```
/**
 * @brief Initialises MotionFX algorithm
 * @param None
 * @retval None
 */
void MotionFX_manager_init(void)
{
    char acc_orientation[4];
    char gyro_orientation[4];
    char mag_orientation[4];

    acc_orientation[0] = 'n';
    acc_orientation[1] = 'w';
    acc_orientation[2] = 'u';

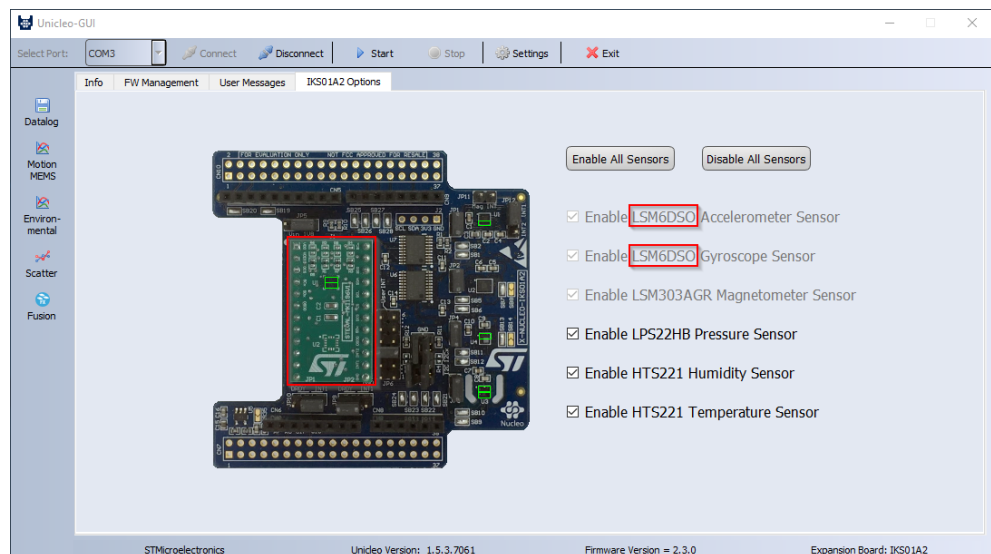
    gyro_orientation[0] = 'n';
    gyro_orientation[1] = 'w';
    gyro_orientation[2] = 'u';

    mag_orientation[0] = 'n';
    mag_orientation[1] = 'e';
    mag_orientation[2] = 'u';
}
```

Note: In this example, no change is needed as the **LSM6DSL** sensor on the X-NUCLEO expansion board and the **LSM6DSO** sensor on the DIL 24 module have the same orientation.

- Step 5.** Build the application and run the **Unicelo-GUI** SW to check if the **LSM6DSO** sensor is active and the Motion Sensor Fusion algorithm is working properly.

Figure 3. Running DataLogFusion application with LSM6DSO sensor (DIL 24 module) in Unicelo-GUI SW



RELATED LINKS

[STEVAL-MK1196V1 evaluation board](#)

[LSM6DSO datasheet freely available at \[www.st.com\]\(http://www.st.com\)](#)

[LSM6DSO application note: AN5192, "LSM6DSO: always-on 3D accelerometer and 3D gyroscope", freely available at \[www.st.com\]\(http://www.st.com\)](#)

Revision history

Table 2. Document revision history

Date	Revision	Changes
11-Jun-2019	1	Initial release

Contents

1	Step by step procedure to replace an on-board sensor with a DIL24 sensor.....	2
	Revision history	6

List of figures

Figure 1.	Add LSM6DSO sensor drivers in a project for the SW4STM32 IDE	2
Figure 2.	Add LSM6DSO sensor including path to a project for the SW4STM32 IDE	3
Figure 3.	Running DataLogFusion application with LSM6DSO sensor (DIL 24 module) in Unicleo-GUI SW	5

List of tables

Table 1.	Unicleo-GUI supported sensors and IDs	4
Table 2.	Document revision history	6

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, please refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2019 STMicroelectronics – All rights reserved