Getting started with CY8CKIT-062S2-AI: PSOC™ 6 AI Evaluation Kit

1 Introduction

The PSOC™ 6 AI Evaluation Kit is a cost-effective, small form-factor development kit. It provides the best of Infineon's solutions to drive adoption of DEEPCRAFT™ Studio for different use cases of machine learning and Infineon's software products.

This kit features a PSOC™ 6 MCU, an AIROC™ CYW43439 Wi-Fi/Bluetooth® combo module, a 512 Mb NOR flash, an onboard programmer/debugger (KitProg3), a PDM-PCM digital microphone interface, a Full-Speed USB device, two user LEDs, and one user button. The board supports operating voltages from 1.8 V to 3.3 V for the PSOC™ 6 MCU.

The kit provides various sensors such as 6-axis motion sensor (BMI270), magnetometer (BMM350), barometric pressure sensor (DPS368), and RADAR sensor (BGT60TR13C) for data collection and creating the machine learning models.

For more information, see CY8CKIT-062S2-AI.

1.1 Kit contents

The kit includes the following:

- PSOC™ 6 AI Evaluation Board
- Inlay card (a printed QR code points to a getting started webpage)

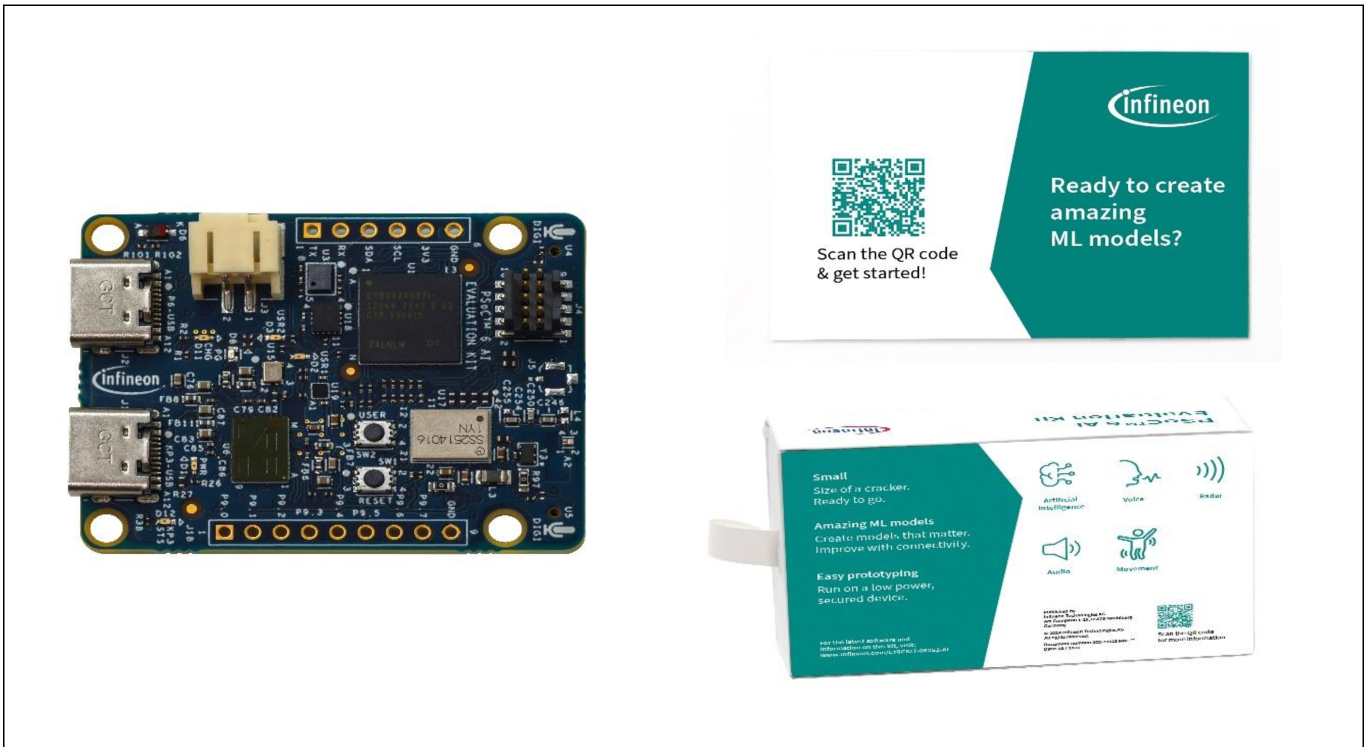

Figure 1 Kit contents

1.2 Requirements

The following software is required to get started:

- ModusToolbox™ software v3.4 or later (for developing PSOC™ 6 MCU-based applications)
- DEEPCRAFT™ v5.3 (for developing machine learning models)
- UART terminal software such as Tera Term or Minicom

2 Out-of-box (OOB) application

The PSOC™ 6 AI Evaluation Kit comes pre-programmed with streaming firmware to stream sensor data from the USB port into DEEPCRAFT™ Studio for labeling and machine learning model creation. The streaming firmware is designed to collect data from all the sensors integrated into the kit, including the microphone, accelerometer, gyroscope, magnetometer, pressure sensor, and radar.

Infineon has developed a new streaming firmware that uses Tensor Streaming Protocol Version 2, offering improved functionality. It is recommended to flash the kits with the new streaming firmware to take advantage of these enhancements. See Tensor Streaming Protocol for real-time data collection to know more about the protocol.

Note: *PSOC™ 6 AI Evaluation kits manufactured before February 2025 are utilizing the old streaming firmware (implemented using protocol version 1); see the manufacturing date printed on the back of your kit box. See Flashing the streaming firmware to learn how to flash the new streaming firmware that uses protocol version 2 onto the kit.*

If you have flashed the PSOC™ 6 AI Evaluation Kit with the new streaming firmware, follow the instructions in Real-time data streaming using the new streaming firmware to stream data into DEEPCRAFT™ Studio.

2.1 Flashing the streaming firmware

To flash the streaming firmware onto the kit, follow the pre-requisites and steps outlined below:

2.1.1 Pre-requisites

- Streaming Firmware - PSOC™ 6 AI Evaluation Kit (hex file)
- Download and install ModusToolbox™ Programmer (flash utility)
- PSOC™ 6 AI Evaluation Kit (CY8CKIT-062S2-AI)

2.1.2 Flash the streaming firmware

1. Download and unzip the hex file to flash the streaming firmware onto the board.
2. Connect the KitProg3 USB connector (J1) port on the board with the PC using the USB cable.

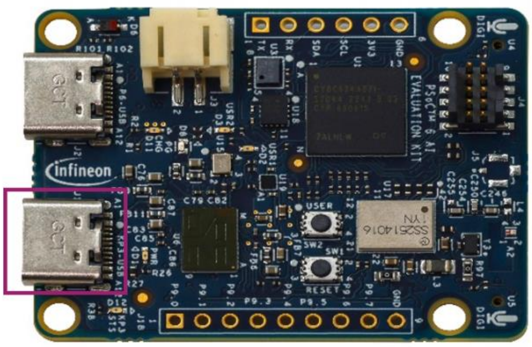

Figure 2 KitProg3 USB connector (J1) port on the board

3. Open **ModusToolbox™ Programmer** from the Windows Start menu.

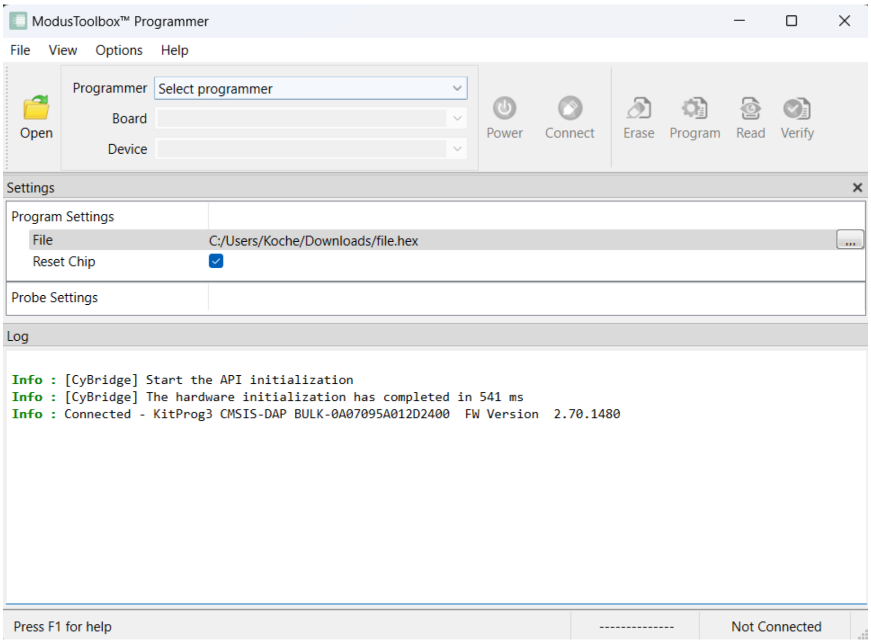

Figure 3 ModusToolbox™ Programmer window

4. In the **Programmer** drop-down box, select the device type as **KitProg3 CMSIS-DAP BULK-XXXXXXXXXXXXXXXX.**
5. In the **Board** drop-down box, select the board type as **CY8CKIT-062S2-AI**.

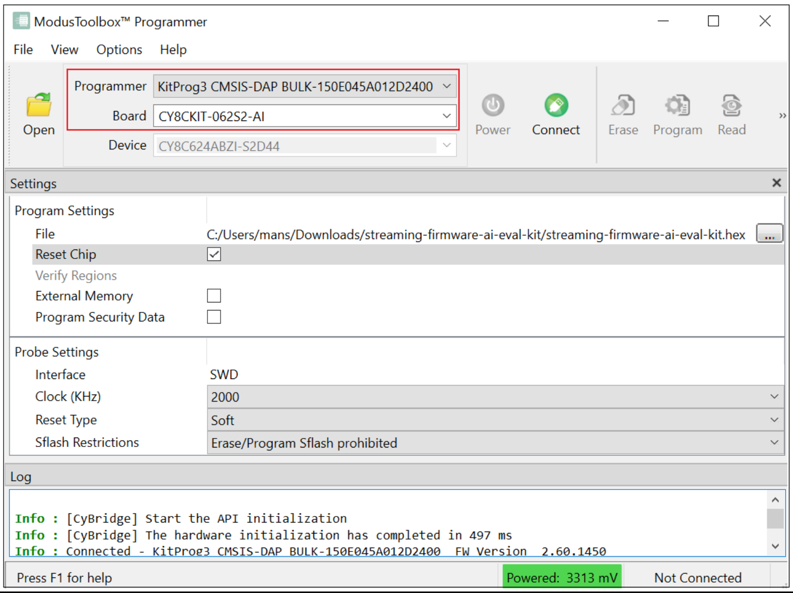

Figure 4 Select programmer and board

6. Click **Open** and navigate to select the streaming firmware (hex file) downloaded earlier.
7. Click **Connect** to establish a connection between the board and the ModusToolbox™ Programmer.

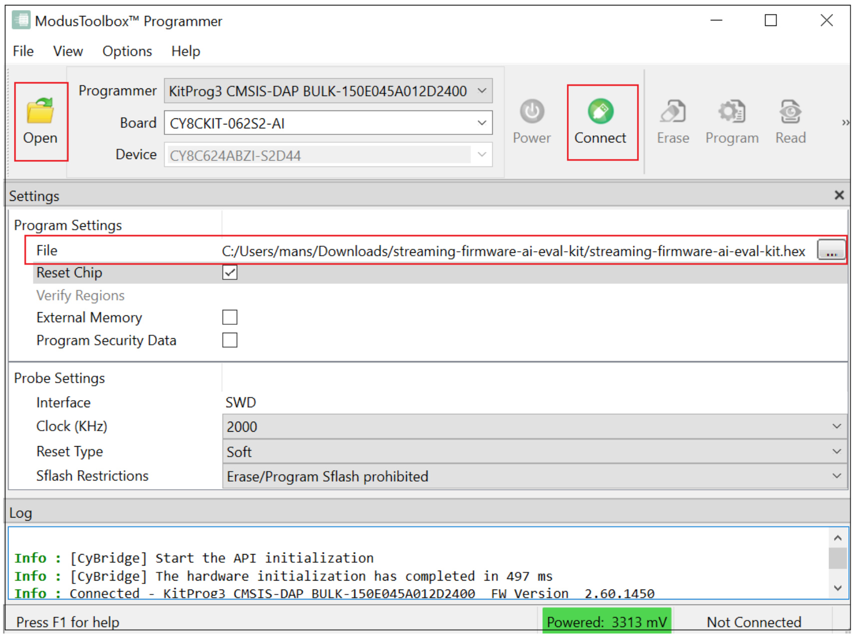

Figure 5 Select the downloaded streaming firmware (hex file)

8. Click **Program** to flash the streaming firmware on the board.

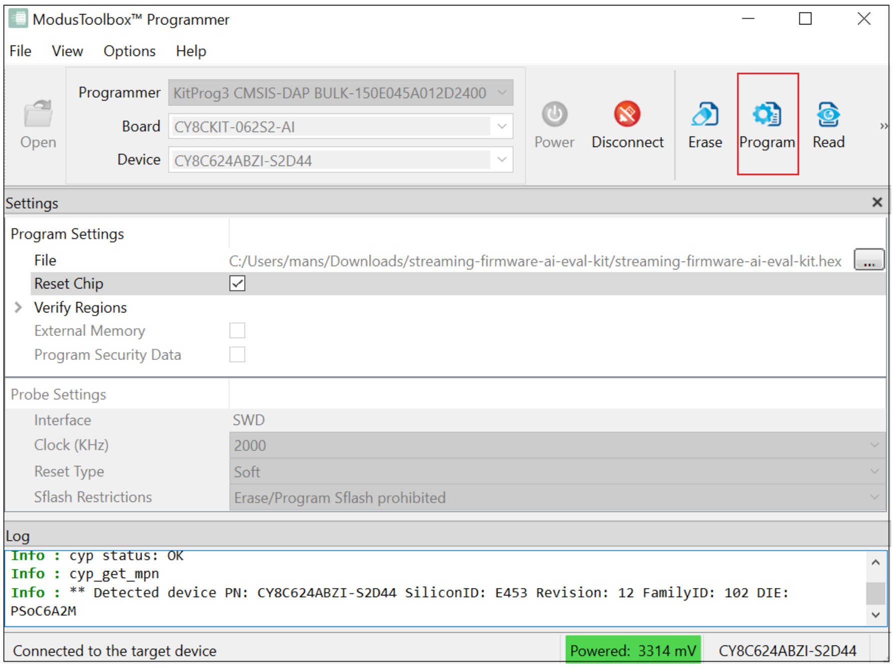

Figure 6 Flash the streaming firmware on the board

9. Ensure that the kit is programmed successfully.

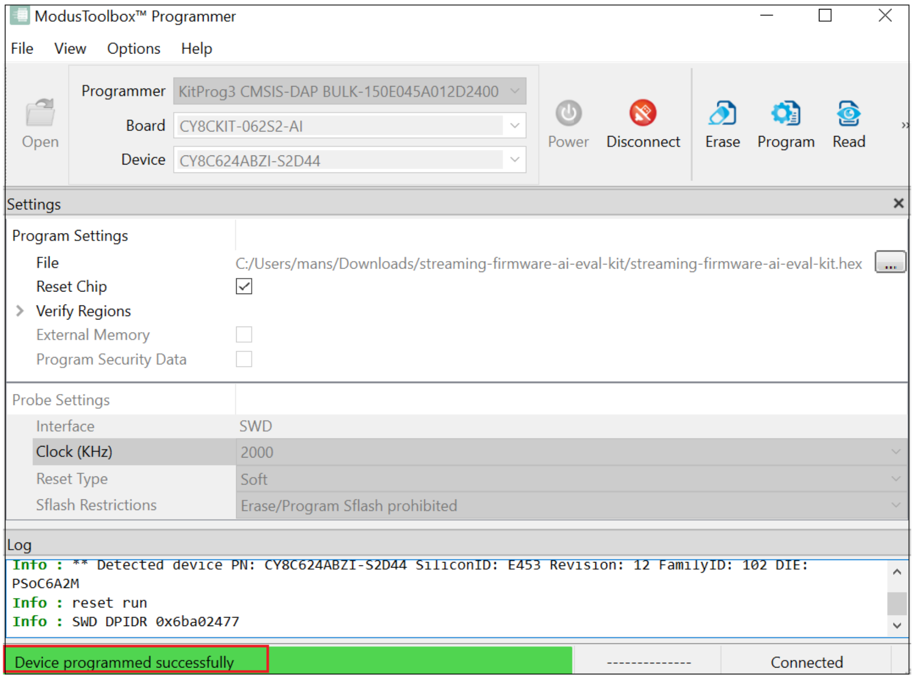

Figure 7 Kit is programmed successfully

10. Now, **disconnect** the KitProg3 USB connector (J1) port.

2.2 Real-time data streaming using the new streaming firmware

By implementing your custom firmware using Protocol version 2, you can follow the guidelines in this topic to collect data from any sensor or development kit in a similar manner. See Tensor Streaming Protocol for real-time data collection to know more about the protocol.
This topic provides information on how to collect data from various sensors present in the PSOC™ 6 AI Evaluation Kit into DEEPCRAFT™ Studio. You can add additional data to your existing projects or build your own dataset from scratch. You can collect both labeled and unlabeled data, depending on your project requirements.

2.3 What type of data can be collected using the PSOC™ 6 AI Evaluation Kit?

After you have flashed the streaming firmware, you can collect various types of data from multiple sensors, including:

- **Audio data**: Pulse density modulation (PDM), and pulse code modulation (PCM) audio data can be collected from the microphone at sample rates of 8 kHz or 16 kHz.
- **Inertial measurement unit (IMU) data**: Data from the six-axis IMU (accelerometer and gyroscope) can be sampled at frequencies of 50 Hz, 100 Hz, 200 Hz, or 400 Hz. This allows for acceleration and gyroscopic angular rate sensing in each spatial direction. The IMU sensor can be configured to collect data from only the accelerometer, only the gyroscope, or both.
- **Magnetometer data**: The three-axis magnetometer provides data at a sample rate of 50 Hz, 100 Hz, 200 Hz, or 400 Hz, enabling geomagnetic field direction and strength sensing.

- **Barometric pressure data**: Data from the barometric pressure sensor can be sampled at a frequency of 8 Hz, 16 Hz, 32 Hz, 64 Hz, or 128 Hz, capturing changes in atmospheric pressure.
- **Radar data**: Radar data can be utilized for various applications, including macro presence detection, micro presence detection, gesture recognition, and other use cases.

2.4 How to collect and label real-time data from the sensors?

Collecting and labeling data in real time is straightforward using Graph UX in DEEPCRAFT™ Studio. Do the following to efficiently collect and label your data.

2.4.1 Connecting PSOC™ 6 AI Evaluation Kit to PC/laptop

Connect the PSOC™ 6 AI Evaluation Kit to the PC or laptop through the PSOC™ 6 USB connector (J2) using a Type-C USB cable.

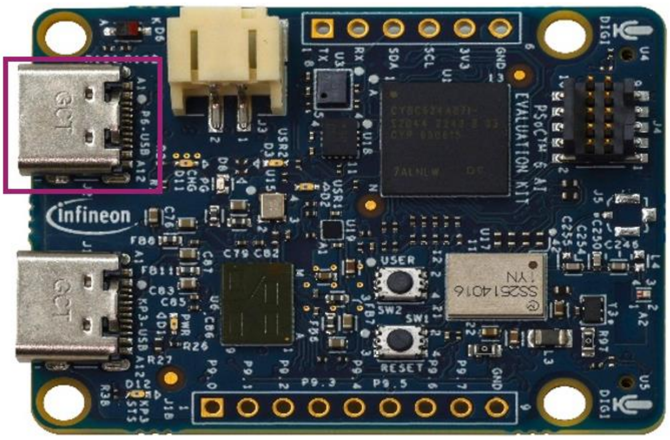

Figure 8 USB connector (J2) port on the board

After connecting the kit, launch the DEEPCRAFT™ Studio and navigate to **Node Explorer Window** > **Library** > **Boards** to check if the board is connected properly. The PSOC™ 6 AI Evaluation Kit, along with all the sensors, should be displayed under the **Boards** unit.

Note: If the ***Node Explorer window*** *is not open, navigate to* ***View*** *>* ***Node Explorer*** *to access the window.*

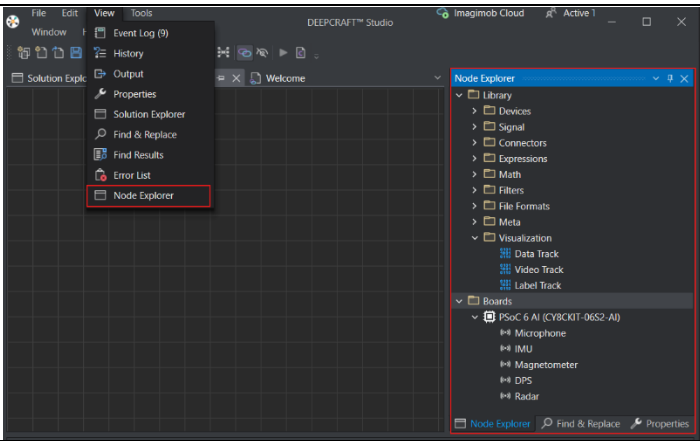

Figure 9 Node Explorer window

2.4.2 Create a Graph UX project

To create an empty Graph UX project, see How to get started with Graph UX?. If you select **Live data collection starter Graph UX** project to collect data, set up the input node and start collecting data. However, if you select **Empty Graph UX** project to collect data, you need to create the data collection graph. The following sections describes how to create the data collection graph.

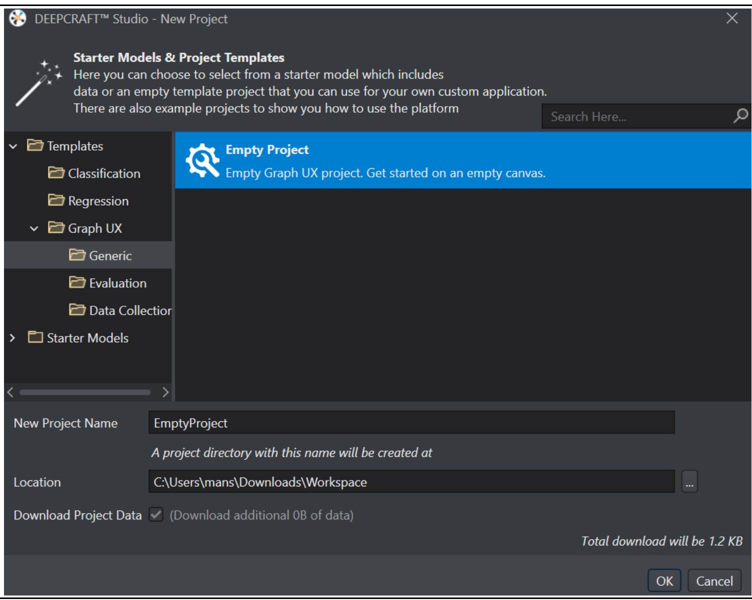

Figure 10 Create empty Graph UX project

2.4.3 Create data collection and data labelling graph

To create a simple version of the data collection graph, you need an input node and a visualization node to view the data from the input node.

2.4.3.1 Select the input node to stream data

Input nodes are designed to collect data from various sensors that are present on the kit, such as a microphone, IMU (accelerometer or gyroscope), magnetometer, pressure sensor, and radar sensor that pass on the data to other nodes in the graph. The input nodes are placed at the beginning of the graph. You can collect data using multiple input nodes simultaneously.

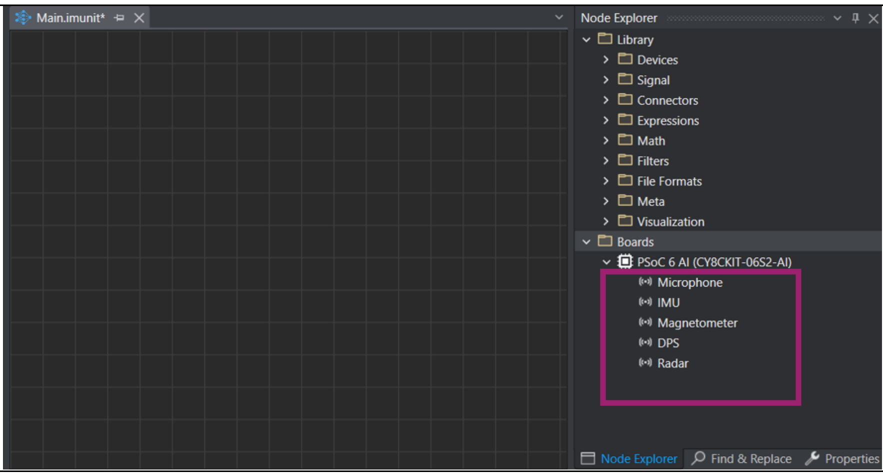

Figure 11 Select input node

- **Microphone node** for streaming audio data
- **IMU node** for streaming accelerometer and gyroscope data
- **Magnetometer node** for streaming magnetometer data
- **Pressure sensor node** for streaming barometric pressure data
- **Radar node** for streaming radar data

Note: If you want to stream video data alongside sensor data, you can use the ***Library*** *>* ***Devices*** *>* ***Local Camera*** *node to stream video using the built-in PC camera. This is particularly useful for labeling data, especially when collecting data from the IMU. To know how to set up the* ***Local Camera*** *node, navigate to the* ***Setting up the Local Camera*** *section here.*

2.4.3.2 Set-up the visualization nodes

The Visualization nodes help in visualizing the data collected from the input nodes as tracks in the session file.

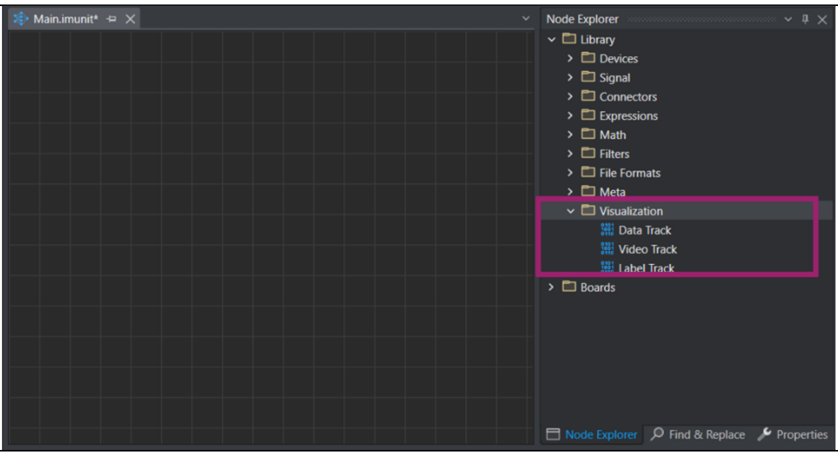

Figure 12 Set-up Visualization nodes

- **Data Track** for visualizing the sensor data collected from Microphone, IMU, DPS, Magnetometer, and Radar node
- **Label Track** for visualizing and labeling the collected data
- **Video Track** for visualizing video data collected from the Local Camera node (applicable only if you set the Local Camera node to collect video data)

Depending on the type of data you want to collect and how you plan to collect it, you can create different data collection graphs.

2.4.4 Real-time data collection and data labeling

2.4.4.1 Setting up the sensors

1. Expand **EmptyProject** directory and double-click the **Main.imunit** to open the canvas.

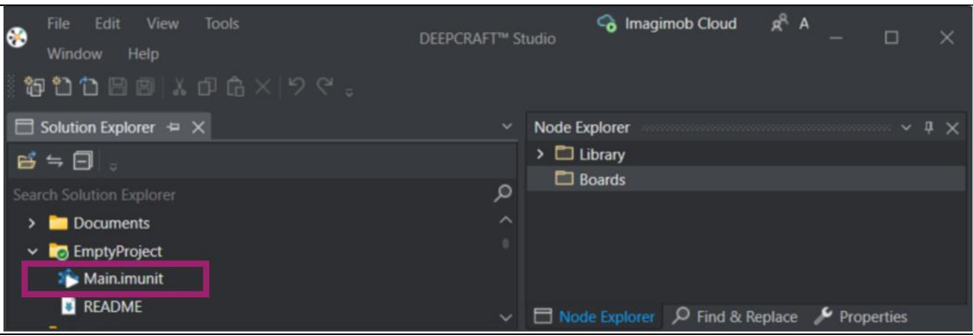

Figure 13 Open the canvas

2. Navigate to the **Node Explorer** window, expand **Library** > **Boards** > **PSOC™ 6 AI (CY8CKIT-062S2-AI)**, and drag and drop the Microphone Sensor, IMU Sensor, Magnetometer, and DSP Sensor onto the canvas.

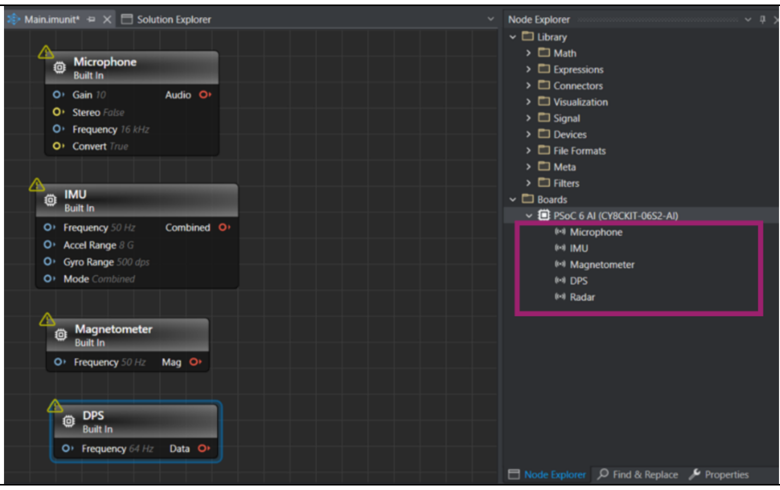

Figure 14 Drag and drop sensors

3. Navigate to the **Node Explorer** window, expand **Library** > **Visualization**, and drag and drop the Data Tracks for each sensor onto the canvas.

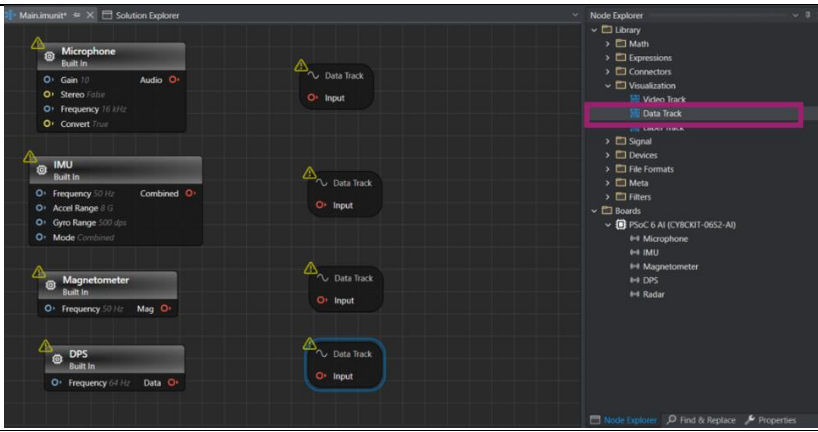

Figure 15 Drag and drop Data Track

4. **Click** on the red icon in the sensor node and drag it over to the red icon in the corresponding Data Track node. This creates a connection between the two nodes.

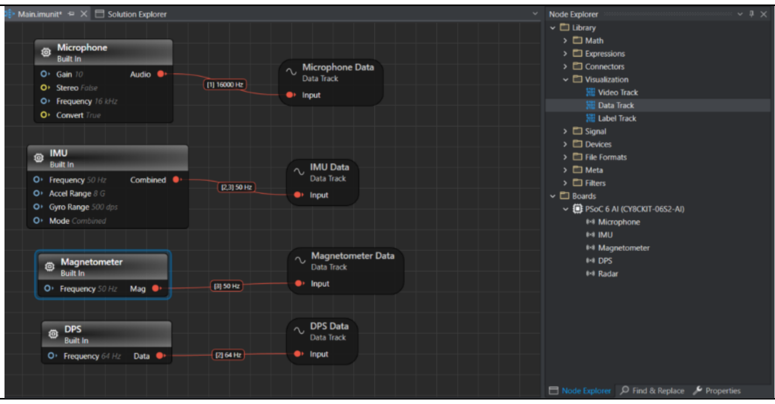

Figure 16 Connect the input node to the Data Track

5. Click on the sensor node to edit the settings in the **Properties** window.

2.4.4.2 Setting up the Predefined Labels

1. Navigate to the **Node Explorer** window, expand **Library** > **Meta**, and drag and drop the **Predefined Labels** unit onto the canvas. The Predefined Labels unit displays the default classes.
2. Define the classes in the **Predefined Labels** node by entering every class in a new line.

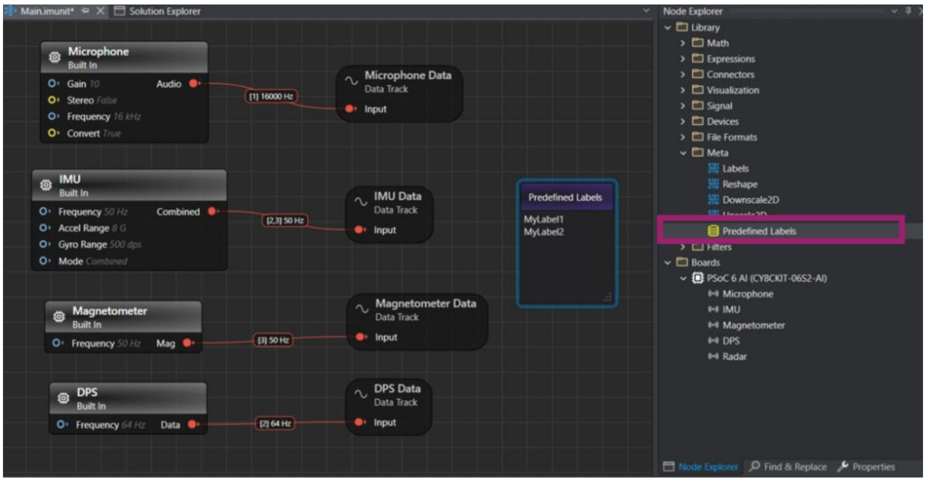

Figure 17 Set-up the Predefined Labels

3. After defining the classes, start collecting and labeling data simultaneously.

2.4.4.3 Data collection and labelling

1. Navigate to the toolbar and click **Start** to open the session file (live.imsession). An empty session file opens, displaying the pre-defined classes in the **Labels** bar.
2. Click **Record** to start capturing real-time data.

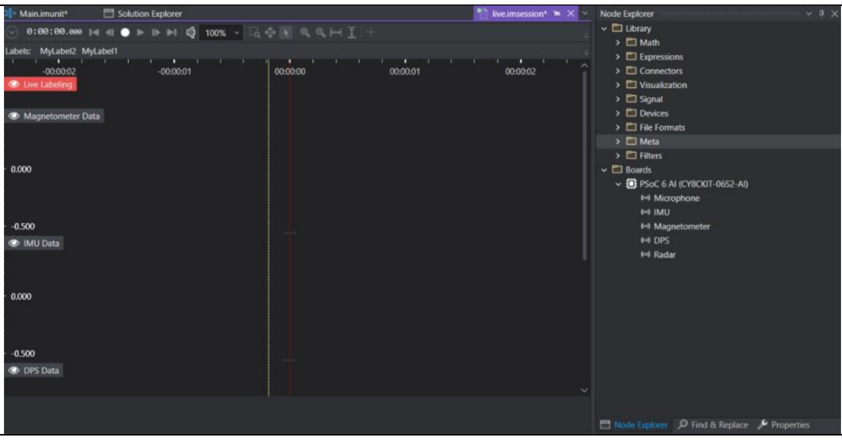

Figure 18 Live session window

3. In the **Labels bar**, select a class to start adding labels to the streaming data.

Figure 19 Label live data

4. Deselect the same class to stop adding the labels. Similarly, you can enable or disable the other classes and label the data.
5. Click **Record** again to stop collecting the data.
6. In **Location**, click the three dots and select the desired location to save the files.
7. In **Session Name**, enter the name of the session file.
8. Under **Track Options**, set the following.

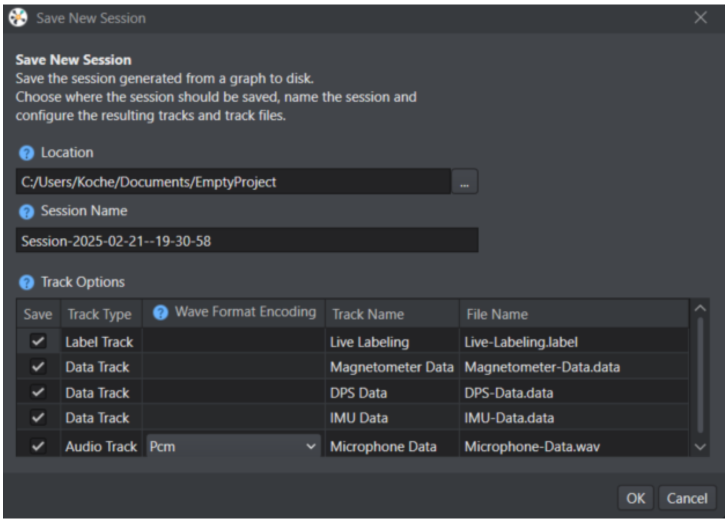

Figure 20 Save new session

9. Select the save checkbox corresponding to the tracks you want to save.
10. Click **OK** to save.
11. Repeat the instructions from Step 2 to collect and save data in multiple session files.

After collecting the data, add data to the required project and start with the machine learning workflow in DEEPCRAFT™ Studio. See Create project and Add data to project to know more.

Note: If you prefer not to collect and label data simultaneously, you can first collect the data. Once the data collection is complete, simply press the ***play*** *button in the session and begin labeling by selecting the appropriate labels in the Labels bar.*

To collect data, connect the respective node to the data track. For more information on setting up and collecting data using the individual nodes, see the following.

- **Microphone data (Microphone node)**: Collecting and labeling data using Microphone sensor
- **IMU sensor data (IMU node)**: Collecting and labeling data using IMU sensor
- **Magnetometer sensor data (Magnetometer node)**: Collecting and labeling data using Magnetometer sensor
- **Barometric pressure sensor (DPS node)**: Collecting and labeling data using Barometric Pressure sensor
- **Radar sensor (Radar node)**: Collecting and labeling data using Radar sensor

Note: You can collect data concurrently from the microphone, IMU, magnetometer, and pressure sensor. However, you must collect data from the Radar sensor independently, without concurrent data collection from any other sensors. To know how to collect data using multiple sensors, see Collecting and Labeling data using multiple sensors.

3 Supported code examples

Code example	Description
mtb-example-ml-deepcraft-deploy-ready-model	This code example demonstrates how to integrate a ready model library from DEEPCRAFT™ Studio on ModusToolbox™. The code example includes six different models such as baby-cry detection, cough detection, alarm detection, siren detection, snoring detection, and hand gesture detection. See README for more details.
mtb-example-ml-deepcraft-deploy-motion	This code example demonstrates how to deploy a Machine Learning model from DEEPCRAFT™ Studio on a PSOC™ 6-family microcontroller. In this specific example, the model uses IMU sensor data to detect gestures. See README for more details.
mtb-example-ml-deepcraft-deploy-audio	This code example demonstrates how to deploy a Machine Learning model from DEEPCRAFT™ Studio on a PSOC™ 6-family microcontroller. In this specific example, the model used is an acoustic model/keyword spotter. See README for more details.

Revision history

Document revision	Date	Description of changes
**	2024-05-27	Initial release.
*A	2025-03-03	Updated the document according to data streaming protocol version 2. Replaced “Imagimob Studio” references with “DEEPCRAFT™ Studio” in the document. Updated Out-of-box (OOB) application section to use the DEEPCRAFT™ Studio workflow. Added Supported code examples section.

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc., and any use of such marks by Infineon is under license.

PSOC™, formerly known as PSoC™, is a trademark of Infineon Technologies. Any references to PSoC™ in this document or others shall be deemed to refer to PSOC™.

Edition 2025-03-03

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email:

erratum@infineon.com

Document reference

002-40023 Rev. *A