



IQS7221E Arduino Example Code



Table of Contents

Introduction	1
Arduino Code Configuration	2
Example Code Flow Diagram	3
SparkFun Board Library Installation	4
Install Encoder and Mouse Libraries	7
Serial Communication and Interface	9



Introduction

This Arduino example code demonstrates how to set up and use the IQS7221E Hall encoder using either I2C or the quadrature pins. The Arduino can also be configured to emulate a mouse scroll wheel using the [Mouse](#) library by providing mouse scroll commands over HID. This example code is specifically aimed at the IQS7221E Evaluation Kit (PCB number AZP1276A2).

The following external Arduino libraries are needed to compile the example pack:

- [Encoder](#): by Paul Stoffregen version 1.4.2.
- [Mouse](#)

This requires an Arduino board that supports HID over USB and 3.3 V logic, such as [Sparkfun's Pro Micro \(3.3V, 8 MHz\)](#).



Arduino Code Configuration

The behaviour and pin assignments of the Arduino code can be configured with the `#define` settings at the start of `iqs7221e-example-code.ino`.

Change the following pin assignments and parameters to suit your hardware:

```
/** Defines */
#define DEMO_IQS7221E_ADDR      0x56
#define DEMO_IQS7221E_POWER_PIN 4
#define DEMO_IQS7221E_RDY_PIN  7

/* Encoder Input Pins (needs to be interrupt-capable pins) */
#define DEMO_ENC_PIN_A          0
#define DEMO_ENC_PIN_B          1

/* UI Selection */
#define DEMO_I2C_OR_ENCODER     0 // 0 - I2C, 1 - Encoder
#define DEMO_HID_ON              1 // Turn on HID and give mouse scroll commands over
USB.
```

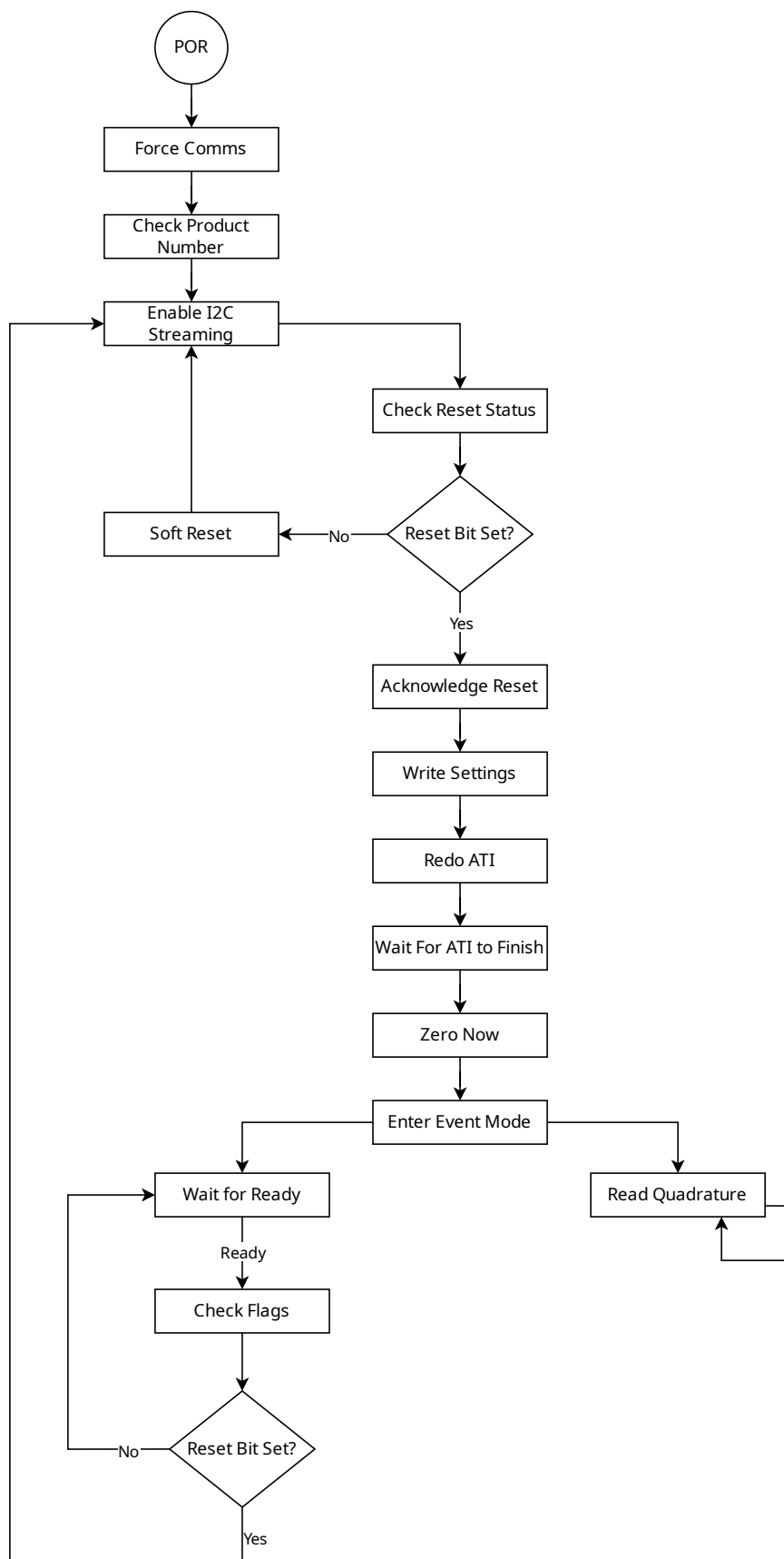
- `DEMO_IQS7221E_POWER_PIN` can be used to power the IQS7221E directly from an Arduino GPIO. This parameter sets which pin to use. This is an optional setting and can be removed if the IQS7221E is powered from the VCC pin or an external power supply.
- `DEMO_IQS7221E_RDY_PIN` sets the pin assignment for the IQS7221E ready pin. This must support external interrupts. On the SparkFun Pro Micro, pins 0, 1, 2, 3, and 7 support interrupts.
- `DEMO_I2C_OR_ENCODER` determines whether the Arduino reads the current encoder interval from the IQS7221E over I2C or from the quadrature pins. Using the quadrature pin requires the [Encoder](#) library.
- `DEMO_HID_ON` sends mouse scroll events over USB HID to the computer to emulate a mouse wheel. Enabling this requires the [Mouse](#) library.
- `DEMO_ENC_PIN_A` and `DEMO_ENC_PIN_B` sets the pin assignments for the IQS7221E quadrature pins, and allows the Arduino to read the encoder interval using the [Encoder](#) library. These pins must be interrupt-capable. On the SparkFun Pro Micro, pins 0, 1, 2, 3, and 7 support interrupts.



Please note that powering an IQS device directly from a GPIO is *generally* not recommended. However, the `DEMO_IQS7221E_POWER_PIN` in this example could be used as an enable input to a voltage regulator.



Example Code Flow Diagram





SparkFun Board Library Installation

To use the SparkFun Pro Micro, the SparkFun Board Library must be installed in the Arduino IDE.

Add the SparkFun Board Library by opening Preferences (**File > Preferences**), and paste the following URL into the "Additional Board Manager URLs" text box.

```
https://raw.githubusercontent.com/sparkfun/Arduino_Boards/master/IDE_Board_Manager/package_sparkfun_index.json
```

Preferences

Settings Network

Sketchbook location: c:\Azoteq\Arduino **BROWSE**

☐ Show files inside Sketches

Editor font size: 14

Interface scale: ☒ Automatic 100 %

Theme: Light (Arduino)

Language: English (Reload required)

Show verbose output during ☐ compile ☐ upload

Compiler warnings: None

☐ Verify code after upload

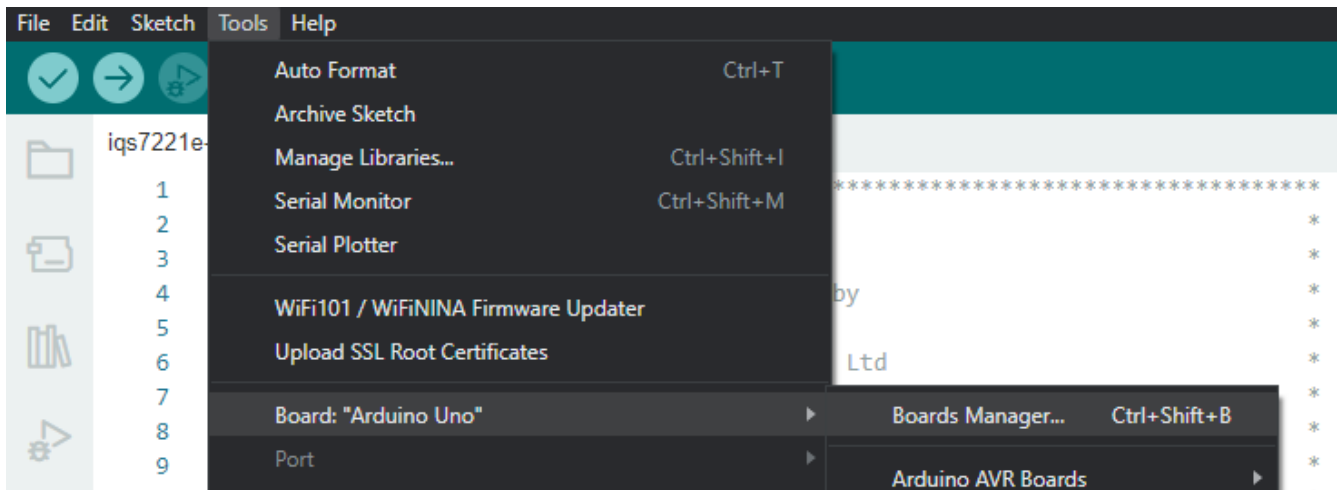
☒ Auto save

☐ Editor Quick Suggestions

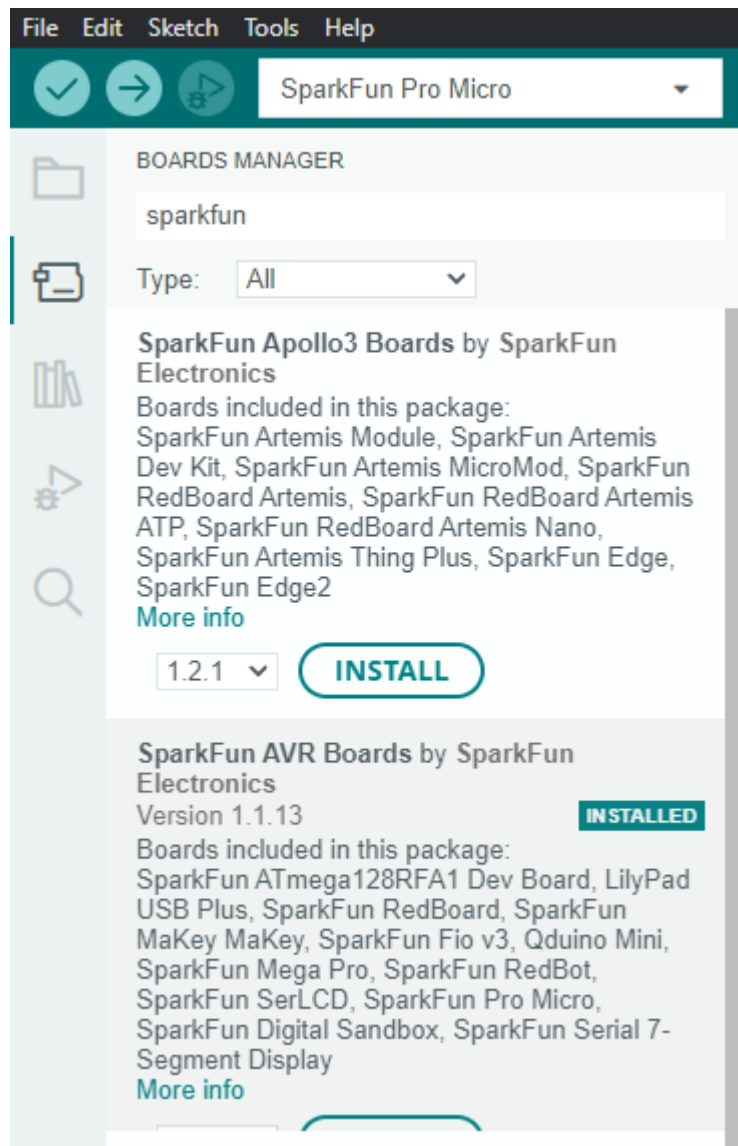
Additional boards manager URLs: https://raw.githubusercontent.com/sparkfun/Arduino_Boards/master/IDE_Board_ **+**

CANCEL **OK**

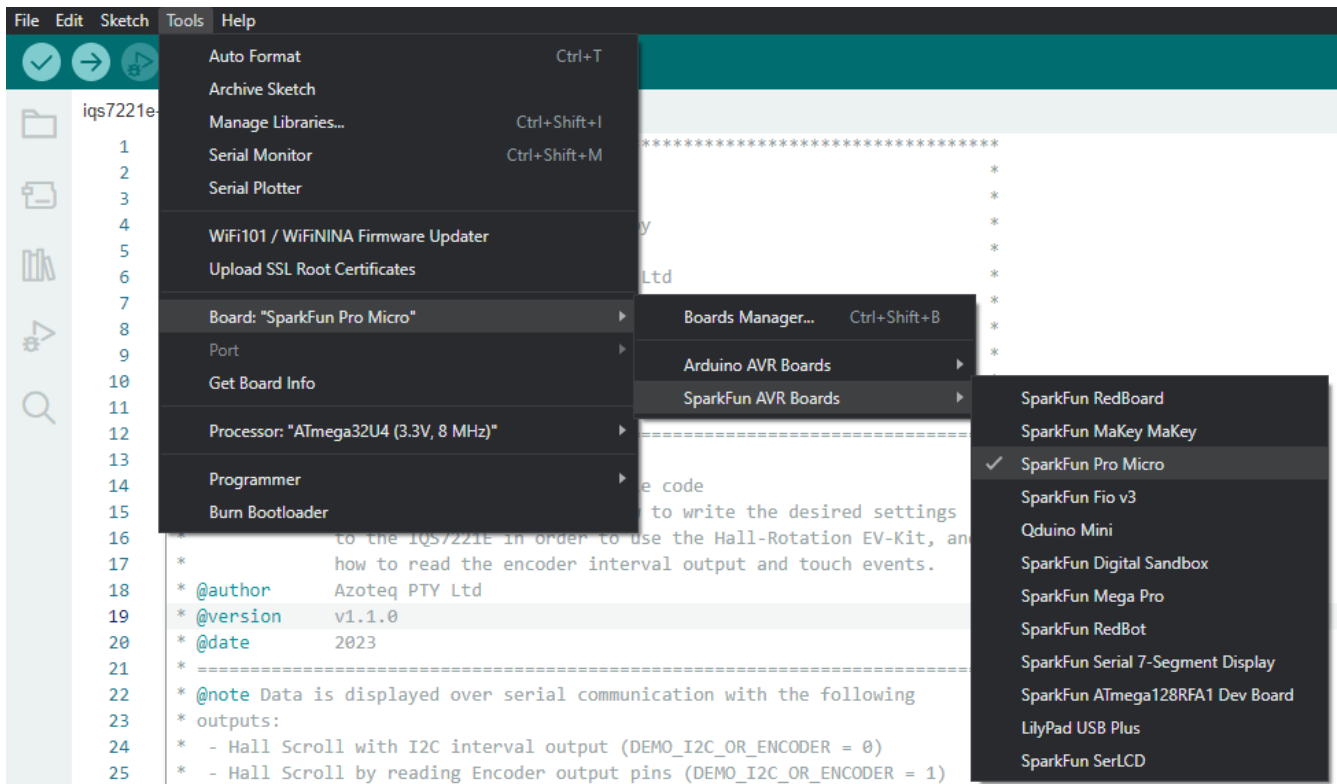
Click "OK". Then open the Board Manager under **Tools > Board > Boards Manager...**



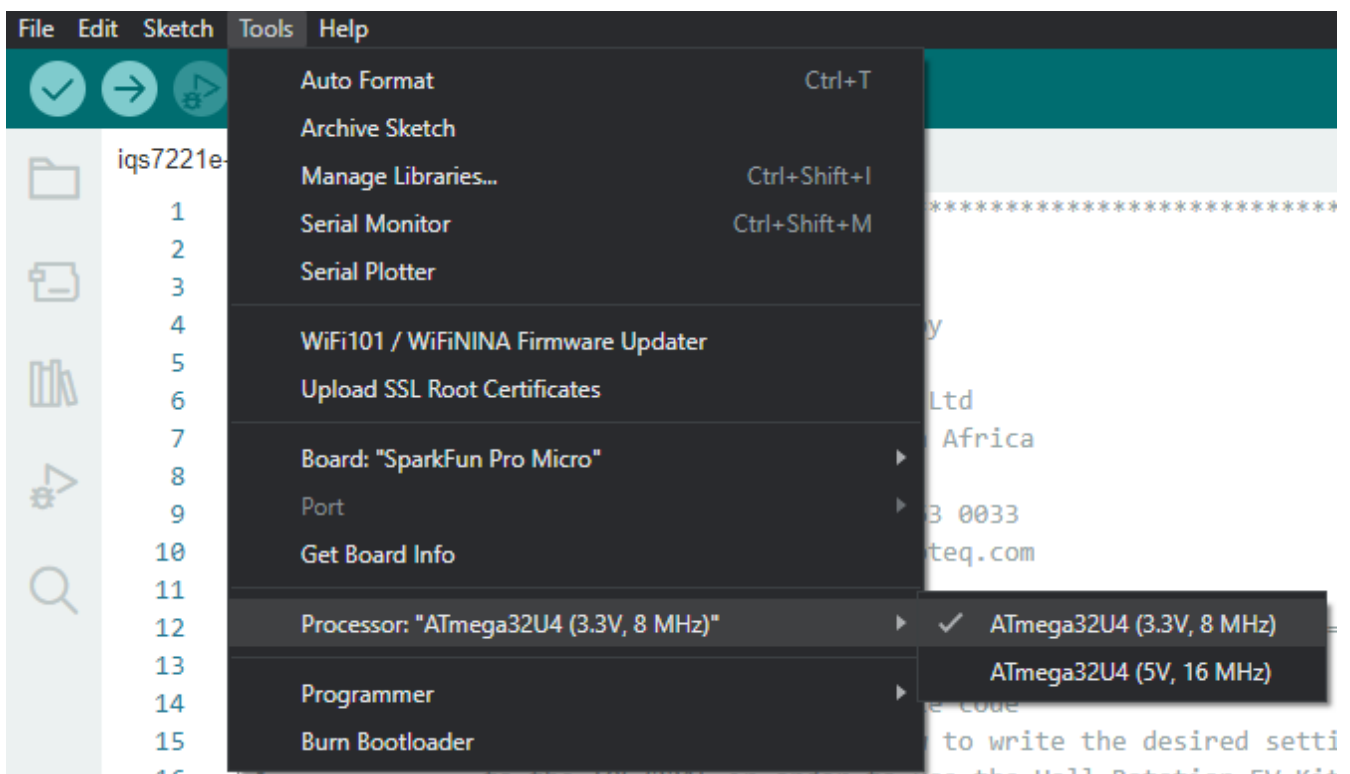
Search for "SparkFun", and install "SparkFun AVR Boards by SparkFun".



You can now select the "SparkFun Pro Micro" in the Board selection menu.



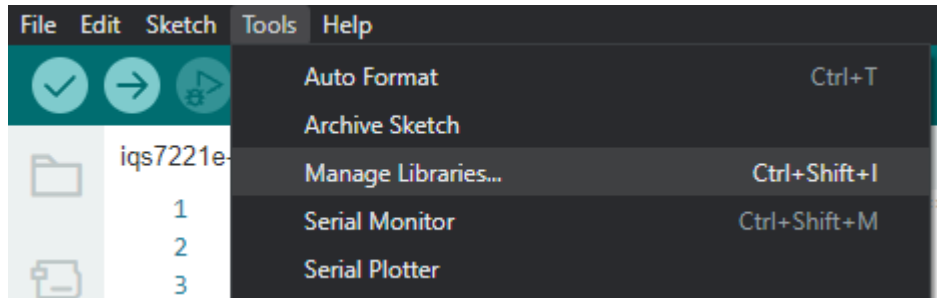
Also be sure to select the "3.3 V, 8 MHz" version under Tools > Processor.



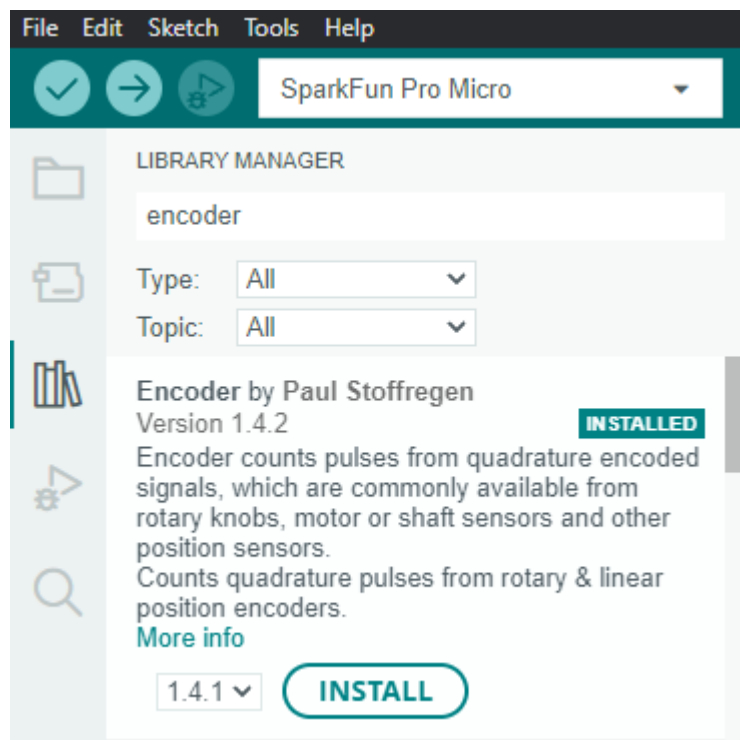
Source: [Pro Micro Hookup Guide](#)

Install Encoder and Mouse Libraries

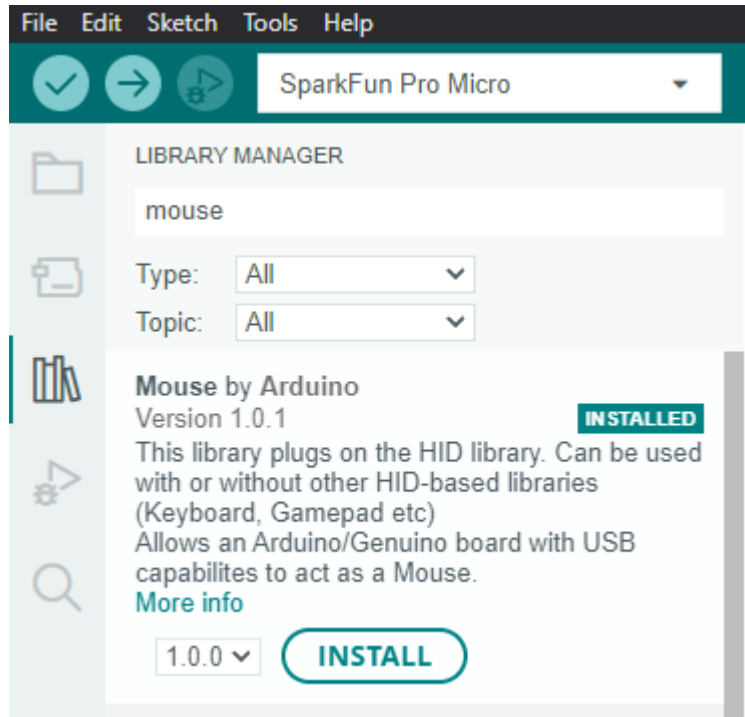
To install the Encoder and Mouse libraries for the example code, open the Library Manager by navigating to **Tools > Manage Libraries...**



Search for "encoder", and install "Encoder by Paul Stoffregen".



Search for "mouse", and install "Mouse by Arduino".





Serial Communication and Interface

The example code provides verbose serial feedback to aid in the demonstration of start-up and operational functions. It also has two built-in commands to demonstrate the IQS7221E device's functionality. To use the built-in commands, the Arduino code simply sends an 'f' or 'r' over the serial interface.

1 - "f\n" - Force open a communication(RDY) window

2 - "r\n" - Request a Software Reset during runtime

It is important to take note of the newline ("\n") character that is needed to complete any serial request. It can be activated in the built-in Arduino IDE serial monitor and is shown inside the blue rectangle in the figure below.

```
Start Serial communication
IQS7221E Ready
IQS7221E Initialization:
  IQS7221E_INIT_VERIFY_PRODUCT
    Product number is: 1283 v1.1
    IQS7221E Release UI Confirmed!
  IQS7221E_INIT_READ_RESET
    Reset event occurred.
  IQS7221E_INIT_UPDATE_SETTINGS
    1. Write System Settings
    2. Write Hall UI Settings
    3. Write Hall Plate Settings
    4. Write Freewheel Settings
    5. Write Hall ATI Settings
    6. Write Button Settings
    7. Write Button Sensor Settings
    8. Write Button Threshold Settings
    9. Write I2C Settings
  IQS7221E_INIT_ACK_RESET
  IQS7221E_INIT_ATI
  IQS7221E_INIT_WAIT_FOR_ATI
    DONE
  IQS7221E_INIT_ZERO
  IQS7221E_INIT_READ_DATA
  IQS7221E_INIT_ACTIVATE_EVENT_MODE
  IQS7221E_INIT_DONE
IQS7221E Initialization complete!
```

Serial Monitor controls: ☒ Autoscroll ☐ Show timestamp **Newline** 115200 baud Clear output