Hochschule Emden/Leer
Department of Mechanical Engineering
Industrial Informatics

University of Applied Sciences
HOCHSCHULE
EMDEN·LEER

GPS Tracking System

Real-Time GPS Tracking with Portenta H7 and IoT GNSS Shield

Manoj Selvaraju

6. Juni 2024

9 Quellen

Introduction I

Overview of Presentation

- **Portenta H7:** A powerful microcontroller board designed for industrial applications.
- **Cat. M1/NB IoT GNSS Shield:** A shield that adds cellular connectivity and GNSS capabilities.
- **Purpose:** To integrate these two components for enhanced IoT solutions.
- **Benefits:**
  - Expanded connectivity options.
  - Real-time geolocation data.
  - Suitable for remote and industrial IoT applications.

About Portenta H7 I

Key Features

- **Processor:** Dual-core ARM Cortex-M7 (480 MHz) and Cortex-M4 (240 MHz).
- **Memory:** 8MB SDRAM, 16MB NOR Flash, 1MB SRAM.
- **Connectivity:** Wi-Fi, Bluetooth, Ethernet, USB, CAN, and more.
- **Industrial Grade:** Suitable for critical and demanding applications.
- **Versatile:** Can be programmed with Arduino IDE.

Portenta H7 I

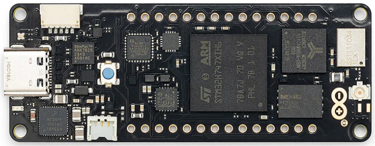


Figure1: Arduino PortentaH7 Top [Ard24c]

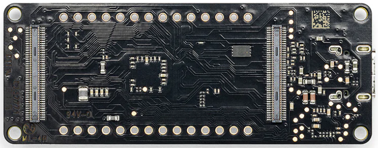


Figure2: PortentaH7 Bottom [Ard24c]

About Cat. M1/NB IoT GNSS Shield I

Key Features

- **Connectivity:** LTE Cat M1, NB-IoT for wide area network communication.
- **GNSS:** Supports GPS, GLONASS, Galileo, BeiDou for precise positioning.
- **Low Power:** Optimized for low power consumption, ideal for battery-operated devices.
- **Applications:** Suitable for asset tracking, remote monitoring, and IoT deployments.

Portenta H7 Cat. M1/NB IoT GNSS Shield I

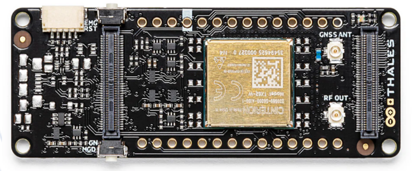


Figure1: Portenta H7 Cat. M1/NB IoT GNSS Shield TopView [Ard24d]

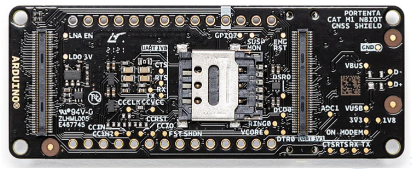


Figure2: Portenta H7 Cat. M1/NB IoT GNSS Shield BottomView [Ard24d]

Benefits of Integration I

Benefits of Integration

- **Enhanced Connectivity:**
  - Multiple network options including cellular and Wi-Fi.
  - Reliable communication in remote areas.
- **Real-time Geolocation:**
  - Accurate tracking of devices in motion.
  - Useful for logistics and fleet management.
- **Low Power Consumption:**
  - Extends battery life for IoT devices.
  - Suitable for remote and long-term deployments.
- **Industrial and Remote Applications:**
  - Monitoring environmental conditions in agriculture.
  - Tracking assets in supply chain management.

Technical Specifications I

Technical Specifications

- **Portenta H7:** [Ard24f]
  - Processor: Dual-core ARM Cortex-M7 (480 MHz) and Cortex-M4 (240 MHz).
  - Memory: 8MB SDRAM, 16MB NOR Flash, 1MB SRAM.
  - Connectivity: Wi-Fi, Bluetooth, Ethernet, USB, CAN, and more.
  - Operating Temperature: -40 to +85 degrees Celsius.
- **Cat. M1/NB IoT GNSS Shield:** [Ard24a]
  - Modem: LTE Cat M1, NB-IoT for IoT applications.
  - GNSS: Supports GPS, GLONASS, Galileo, BeiDou.
  - Power Consumption: Ultra-low power, ideal for battery-operated devices.
  - Antenna: External antenna for improved signal reception.

Integration Process I

Integration Process

Hardware Connection:

- Stack the Cat. M1/NB IoT GNSS Shield on top of the Portenta H7.
- Ensure secure connection of pins and proper alignment.

Software Setup:

- Install necessary libraries in the Arduino IDE.
- Configure settings for cellular and GNSS functionalities.

Writing and Uploading Code:

- Develop code to handle connectivity and data transmission.
- Upload the code to the Portenta H7 and test the integration.

Project: Real-time GPS Tracker I

Sample Project: Real-time GPS Tracker

The goal of this project is to create a real-time GPS tracker using the Portenta H7 microcontroller board and the Cat. M1/NB IoT GNSS Shield. The tracker will capture GPS coordinates (latitude, longitude, and altitude) and send this data over a cellular network to a remote server.

Hardware and Software Requirements:

- Portenta H7
- Portenta Cat. M1/NB IoT GNSS Shield
- A GPS active antenna (e.g GPS active antenna 28dB) connected to the GNS ANT antenna connector on the top side of the shield.
- Connector converter for the active GPS antenna to the board, Coaxial to SMA

Project: Real-time GPS Tracker II

Sample Project: Real-time GPS Tracker

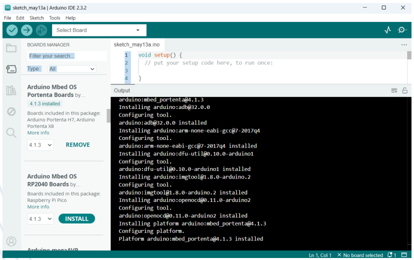
- SIM card (standard pre-paid or post-paid SIM card that supports Cat M1 or NB-IoT connectivity, along with details of APN settings and PIN code usually 0000 or 1234. These are usually provided by your SIM card provider)
- Arduino IDE

Procedure

- Stack the Cat. M1/NB IoT GNSS Shield on the Portenta H7.
- Attach the GNSS antenna to the shield.
- Insert the SIM card with a data plan into the shield.

Project: Real-time GPS Tracker III

Sample Project: Real-time GPS Tracker

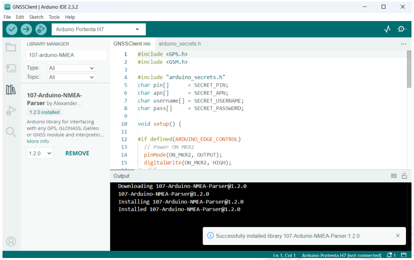

sketch_may13a | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Select Board
BOARDS MANAGER
Filter your search...
Type: All
Arduino Mbed OS Portenta Boards by...
4.1.3 installed
Boards included in this package: Arduino Portenta H7, Arduino Portenta X8
More info
4.1.3
REMOVE
Arduino Mbed OS RP2040 Boards by...
Boards included in this package: Raspberry Pi Pico
More info
4.1.3
INSTALL
sketch_may13a.ino
Output
Ln 1, Col 1
No board selected

```
void setup() {
  // put your setup code here, to run once:

}
```

```
arduino:mbed_portenta@4.1.3
Installing arduino:adb@32.0.0
Configuring tool.
arduino:adb@32.0.0 installed
Installing arduino:arm-none-eabi-gcc@7-2017q4
Configuring tool.
arduino:arm-none-eabi-gcc@7-2017q4 installed
Installing arduino:dfu-util@0.10.0-arduino1
Configuring tool.
arduino:dfu-util@0.10.0-arduino1 installed
Installing arduino:imgtool@1.8.0-arduino.2
Configuring tool.
arduino:imgtool@1.8.0-arduino.2 installed
Installing arduino:openocd@0.11.0-arduino2
Configuring tool.
arduino:openocd@0.11.0-arduino2 installed
Installing platform arduino:mbed_portenta@4.1.3
Configuring platform.
Platform arduino:mbed_portenta@4.1.3 installed
```

Figure1: Mbed Core Installation


GNSSClient | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Arduino Portenta H7
LIBRARY MANAGER
107-arduino-NMEA
Type: All
Topic: All
107-Arduino-NMEA-Parser by Alexander...
1.2.0 installed
Arduino library for interfacing with any GPS, GLONASS, Galileo or GNSS module and interpretin...
More info
1.2.0
REMOVE
GNSSClient.ino
arduino_secrets.h
Output
Ln 1, Col 1
Arduino Portenta H7 [not connected]

```
#include <GPS.h>
#include <GSM.h>

#include "arduino_secrets.h"
char pin[]      = SECRET_PIN;
char apn[]      = SECRET_APN;
char username[] = SECRET_USERNAME;
char pass[]     = SECRET_PASSWORD;

void setup() {

#if defined(ARDUINO_EDGE_CONTROL)
  // Power ON MKR2
  pinMode(ON_MKR2, OUTPUT);
  digitalWrite(ON_MKR2, HIGH);
```

```
Downloading 107-Arduino-NMEA-Parser@1.2.0
107-Arduino-NMEA-Parser@1.2.0
Installing 107-Arduino-NMEA-Parser@1.2.0
Installed 107-Arduino-NMEA-Parser@1.2.0
```

Successfully installed library 107-Arduino-NMEA-Parser:1.2.0

Figure2: NMEA Library Installation

Arduino IDE

Make sure that you have the latest Arduino Mbed OS Portenta core installed. You can install it with the board manager under `Tools > Board > Board Manager` 12. With the core installed and the board selected, navigate to `File > Examples > GSM > GNSSClient`. You will

Project: Real-time GPS Tracker IV

Sample Project: Real-time GPS Tracker

open an sketch that connects to the SIM card provider and initializes the active GPS antenna. At this point, it will print out GPS readings.
Make sure you go to the arduino_secrets.h tab and:

- Add the PIN of the SIM card you are using and store it in the variable `SECRET_PIN`.
- Browse your IT provider and check the mobile APN link, e.g önline.provider.comßave it inside the `SECRET_APN`

After finishing this setup, compile and upload the program. We will see the NMEA data in the Serial Monitor.

GNSS Client

```
#include <GPS.h>
#include <GSM.h>
#include "arduino_secrets.h"
char pin[]      = SECRET_PIN;
char apn[]      = SECRET_APN;
char username[] = SECRET_USERNAME;
char pass[]     = SECRET_PASSWORD;

void setup() {

       #if defined(ARDUINO_EDGE_CONTROL)
       // Power ON MKR2
       pinMode(ON_MKR2, OUTPUT);
       digitalWrite(ON_MKR2, HIGH);
       #endif
       Serial.begin(115200);
       while (!Serial) {}
       // To enable AT Trace debug uncomment the
           following lines
```

GNSS Client

```
            //GSM.trace(Serial);
            //GSM.setTraceLevel(4);
            Serial.println("Starting Carrier Network
                registration");
            if(!GSM.begin(pin, apn, username, pass, CATNB)){
                            Serial.println("The board was
                not able to register to the network...");
                   // do nothing forevermore:
                   while(1);
            }
            Serial.println("\nEnable GNSS Engine...");
            // GPS.begin() start and eanble the GNSS engine
            GPS.begin();
            Serial.println("\nGNSS Engine enabled...");
            }
```

GNSS Client

```
        void loop() {
                // Print out raw NMEA strings.
                // For parsed output look at the
                     MicroNMEA_integration example.
                if(GPS.available()){
                        Serial.print((char) GPS.read());
                        delay(1);
                }
                // After geting valid packet GPS.end() can be used
                      to stop and
                // disable the GNSS engine
                // GPS.end();
        }
```

Parse NMEA GPS Sentences I

- It was not possible to evaluate those messages (NMEA sentences) of GPS data in Serial Monitor. So we need a NMEA Parser to convert messages received from the GPS modem, parsing and saving them into variables.
- In this way, it is possible to interact with the data that you need for your application, for instance getting only latitude and longitude. You will be able to save those values into variables, instead of having the whole NMEA messages.
- Go to `Sketch > Include Library > 107-Arduino-NMEA-Parser library` 12 and Install.
- Open the example from the library at `Examples > 107-Arduino-NMEA-Parser > NMEA-Basic` and modify accordingly

Include the libraries

```
#include "GPS.h"
#include "GSM.h"
#include "ArduinoNmeaParser.h"
#include "Arduino_secrets.h"

char pin[]      = SECRET_PIN;
char apn[]      = SECRET_APN;
char username[] = SECRET_LOGIN;
char pass[]     = SECRET_PASS;
```

Inside the setup() initialize the GSM and GPS modules

```
void setup(){
        Serial.begin(115200);
        while (!Serial) {}
        Serial.println("GSM...");
        GSM.begin(pin, apn, username, pass, CATNB);
        Serial.println("GPS...");
        GPS.begin();
        Serial.println("Success");
}
```

Edit the loop to parse the GPS readings instead of the Serial1

```
void loop(){
        while(GPS.available()){
                parser.encode((char)GPS.read());
        }
}
```

Use Cases and Applications I

Use Cases and Applications

- **Asset Tracking:**
  - Real-time location monitoring.
  - Reduces loss and improves asset management.
- **Remote Monitoring and Control:**
  - Monitor environmental conditions remotely.
  - Control devices and machinery from a distance.
- **Environmental Monitoring:**
  - Track weather conditions, air quality, and other environmental factors.
  - Useful for smart agriculture and urban planning.
- **Smart Agriculture:**
  - Monitor soil moisture, crop health, and livestock tracking.
  - Optimize resource usage and improve yield.
- **Industrial IoT:**
  - Monitor and control industrial processes.
  - Enhance efficiency and reduce downtime.

Quellen I

Bibliography / References

[Ard24a] Arduino. *Arduino Portenta Cat. M1/NB IoT GNSS Shield Documentation*. 2024. URL: https://docs.arduino.cc/hardware/portenta-cat-m1-nb-iot-gnss-shield/.

[Ard24b] Arduino. *Arduino Portenta Cat. M1/NB IoT GNSS Shield MKRGSM Reference*. 2024. URL: https://www.arduino.cc/en/Reference/MKRGSM.

[Ard24c] Arduino. *Arduino Portenta Cat. M1/NB IoT GNSS Shield online store*. 2024. URL: https://store.arduino.cc/products/portenta-h7.

[Ard24d] Arduino. *Arduino Portenta Cat. M1/NB IoT GNSS Shield online store*. 2024. URL: https://store.arduino.cc/products/portenta-catm1/.

University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Quellen II

[Ard24e] Arduino. *Arduino Portenta Cat. M1/NB IoT GNSS Shield Reference MKRGSM Reference*. 2024. URL: https://www.arduino.cc/en/Reference/ArduinoMKRGPS.

[Ard24f] Arduino. *Arduino Portenta H7*. 2024. URL: https://docs.arduino.cc/hardware/portenta-h7/.