

Wearable Running Monitoring System: GPS Tracker with Inertial Correction

Introduction

The project consists of the design and implementation of a wearable embedded system for monitoring running activity, with the primary objective of accurately estimating the distance traveled by the user. Inspired by commercial fitness tracking applications such as Strava, the project aims to evaluate how closely a similar level of functionality and accuracy can be achieved using low-cost hardware and a resource-constrained platform.

A key motivation behind this work is addressing the inherent limitations of standalone GPS-based measurements, particularly the phenomenon of GPS drift, which leads to error accumulation during stationary periods. To mitigate this issue, the system integrates a sensor fusion approach that combines absolute positioning data with inertial measurements, enabling the validation of real user movement and the filtering of spurious displacement.

Beyond its technical objectives, the system is also designed as a practical alternative to smartphone-based tracking, allowing the user to monitor running activity without relying on a mobile device, which may be inconvenient or prone to damage during physical activity. At the same time, the project serves as an exploration of the trade-offs between cost, hardware limitations, and performance, providing insight into how closely a low-budget embedded solution can approximate the behavior of commercial-grade systems.

[GitHub Repository](#)

General Description

The system is composed of multiple hardware and software modules that interact to achieve the final functionality.

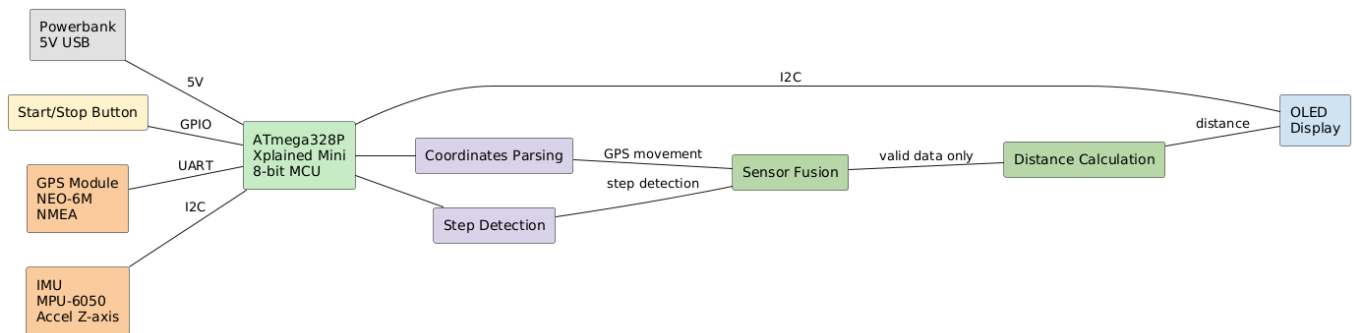
Main modules:

- **Microcontroller (ATmega328P):** coordinates the entire system, processes data, and implements the filtering algorithm.
- **GPS Module (NEO-6M):** provides geographic coordinates in the form of NMEA sentences via UART.
- **Inertial Sensor (MPU-6050):** detects user movement using the accelerometer (Z-axis).
- **OLED Display (SSD1306):** displays information to the user.
- **Power Supply:** 5V power bank for portability.

Module interaction:

1. GPS sends coordinate data to the microcontroller
2. MPU-6050 sends acceleration data
3. The microcontroller processes data and applies the sensor fusion algorithm
4. Distance is updated only if movement is validated
5. Results are displayed on the OLED

A key contribution of this project is a lightweight sensor fusion mechanism that validates GPS-based displacement using inertial step detection. This approach significantly reduces GPS drift without relying on computationally expensive filtering algorithms, making it suitable for resource-constrained microcontrollers.



Hardware Design

List of components:

Component	Model / Specifications
Microcontroller	ATmega328P Xplained Mini
GPS Module	u-blox NEO-6M V2 + 25x25mm ceramic antenna
Inertial Sensor (IMU)	MPU-6050 (Accelerometer + Gyroscope)
Display	0.96" OLED, SSD1306 (I2C)
Power Supply	5V Powerbank
Breadboard	400 points
Wires	10cm DuPont wires
Mounting elements	Spacers / Patafix / pins
Enclosure	3D printed case

Software Design

Description of the firmware:

- development environment (if any) (e.g. AVR Studio, CodeVisionAVR)
- libraries and 3rd-party sources (e.g. Procyon AVRlib)
- algorithms and data structures to be implemented
- (stage 3) implemented sources and functions

Rezultate Obținute

Describe the results obtained after completing the project.

Concluzii

Download

Archive(s) containing all project files: source code, schematics, etc. A README, ChangeLog, and build/upload script are strongly recommended.

Files should be uploaded using **Add Images or other files**. Namespace format: **:pm:prj20??:c?** or **:pm:prj20??:c?:student_name**.

Jurnal

You may include a journal section to track project progress.

Bibliografie/Resurse

List of documents, datasheets, and online resources used, grouped into **Software Resources** and **Hardware Resources** if applicable.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2026/andrei.batasev/vlad.hosu>



Last update: **2026/05/05 11:50**

