Real Time GPS Tracker using Tiva series Microcontroller and SIM808 module

Mohamed Aasil S, Pydimalla Joel, Abhimanyu Valsarajan, Ananthakrishnan N R, Sai Mohnish M

Amrita School Of Computing, Amrita University, Coimbatore, India

Email: cb.en.u4cse21638@cb.students.amrita.edu, , cb.en.u4cse21649@cb.students.amrita.edu, cb.en.u4cse21601@cb.students.amrita.edu, cb.en.u4cse21604@cb.students.amrita.edu, cb.en.u4cse21633@cb.students.amrita.edu,

***Abstract*— The Real-Time GPS Tracker project utilizes the Tiva C Series TM4C123GH6PM microcontroller and SIM808 module for instantaneous location tracking. Leveraging UART communication, the Tiva C Series interfaces with the SIM808 module, sending AT commands for GPS initialization and parsing NMEA sentences to extract real-time latitude and longitude data. This compact and versatile system ensures accurate, on-the-fly tracking, making it ideal for applications like vehicle monitoring or asset tracking. The project showcases the synergy of hardware and software, enabling real-time data processing and the potential for remote monitoring through server communication, providing a robust solution for diverse tracking requirements.**

Keywords—Tiva Board, SIM808, Real-Time Data Transmission, GPS, GNSS.

I. Introduction

A. *Problem Statement*

Design and develop an embedded systems solution for a compact, power-efficient, and real-time GPS tracking device. The device should address the growing demand for efficient location tracking in various applications, such as asset management, personal tracking, and vehicle monitoring. The primary challenges to be addressed include miniaturization to ensure portability, optimization of power consumption for prolonged battery life, and the implementation of a real-time tracking system to provide timely and accurate location updates. The solution should consider the integration of advanced positioning technologies, effective power management strategies, and seamless communication protocols to enable reliable, continuous tracking in diverse environments. The goal is to create a robust, user-friendly GPS tracking device that meets the evolving needs of both consumer and industrial markets.

B. *Objective*

The primary objective of this project is to interface SIM808 module with a Tiva board, a powerful ARM Cortex-M based microcontroller. The data from the module will be collected, processed, and then transmitted to the Tiva board. This project seeks to implement functionalities such as GNSS initialization, GPRS configuration, and HTTP communication to facilitate the transmission of location data to a cloud-based platform.

C. *Scope of the Project*

This project will utilize the SIM808 module, GPS technology, and the Tiva series development board to create a comprehensive GPS tracking device. The SIM808 module, combining GSM/GPRS and GPS functionalities, will serve as the core communication component, facilitating real-time data transmission and reception. The GPS module will provide accurate location information, enhancing the device's positioning capabilities. The Tiva development board will act as the embedded system's processing unit, managing data processing, power management, and communication protocols. Integration of these components will be crucial for achieving the project's objectives, including miniaturization, power efficiency, and real-time tracking, while ensuring seamless communication and reliable operation in various tracking scenarios.

II. Module Selection and Overview

A. *Tiva Series Microcontroller (TM4C123GH6PM)*

The TM4C123GH6PM is a microcontroller from Texas Instruments' Tiva C Series. It features an ARM Cortex-M4F processor with floating-point unit and a variety of peripherals, making it suitable for a wide range of embedded applications. Some key features of the TM4C123GH6PM include a 32-bit ARM Cortex-M4F core running at speeds up to 80 MHz, on-chip memory including Flash and RAM, multiple communication interfaces (UART, I2C, SPI), PWM modules, analog-to-digital converters (ADC), and a variety of timers. It is often used in applications such as industrial control systems, IoT devices, and other embedded systems that require reliable and efficient processing capabilities.

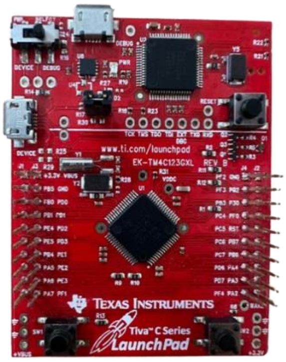

Figure 1. TM4C123GH6PM Microcontroller

B. SIM 808 Module

The SIM808 module is a compact and integrated device designed for communication and positioning applications. It operates on a flexible power supply range of 5-26V, accommodating voltages as low as 9V with a minimum current of 2A. The module supports direct power supply through a 9V port or can be powered by a lithium battery in the range of 3.5-4.2V, making it suitable for portable applications. Equipped with dual SMA antenna ports for GPS and GSM, as well as IPX ports for easy embedding in aluminum boxes, the module offers versatility in antenna connections. With a TTL serial port, it allows seamless switching between GPS and GSM functions. Supporting Quad-band GSM frequencies, GPS precision up to -165dBm, and features such as SMS, GPRS data transmission, and voice calls, the SIM808 is a comprehensive solution for real-time tracking and communication in a variety of applications.

Figure 2. SIM808 Module

III. HARDWARE SETUP

A. Tiva Board Configuration

The core component of the system is the Tiva board, [TM4C123GH6PM], utilized as the central processing unit. The board is configured to interface with the SIM808 module. Power supply connections, clock settings, and pin configurations are established as per the datasheet specifications to ensure proper functionality.

B. SIM808 Interfacing

1) Power Supply:

Connect the VCC pin of the SIM808 module to an appropriate power source (make sure it matches the module's specifications).

Connect the GND pin of the SIM808 module to the ground of your circuit.

2) Serial Communication:

The TM4C123GH6PM and SIM808 communicate through serial communication. Connect the TX (transmit) pin of the TM4C123GH6PM to the RX (receive) pin of the SIM808, and vice versa.

Connect the common ground between TM4C123GH6PM and SIM808.

3) Control Pins:

Some modules may have additional control pins. Connect them based on the module's datasheet. Common control pins include:

RESET: Connect to a GPIO pin on the TM4C123GH6PM for resetting the SIM808.

PWRKEY or POWER: Connect to a GPIO pin for controlling the power state of the SIM808

.

4) Antenna:

Connect the antenna to the designated antenna port on the SIM808 module.

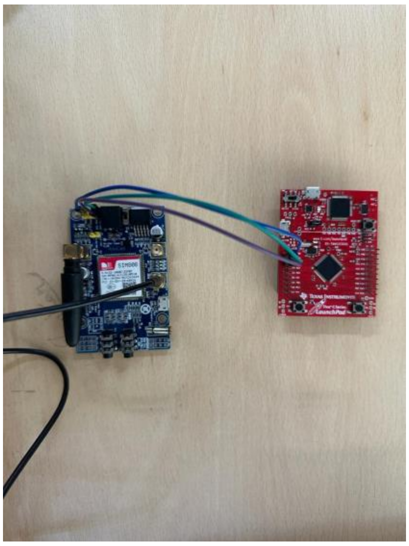

Figure 3. Connections and Setup

IV. SOFTWARE IMPLEMENTATION

A. Importing Header Files

The following header files are imported:

```
#include <stdint.h>
#include <stdbool.h>
#include <string.h>

#include "inc/hw_gpio.h"
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "inc/hw_ints.h"
#include "driverlib/fpu.h"
#include "driverlib/debug.h"
#include "driverlib/sysctl.h"
#include "driverlib/pin_map.h"
#include "driverlib/gpio.h"
#include "driverlib/pwm.h"
#include "driverlib/timer.h"
#include "driverlib/adc.h"

#include "uartstdio.h"
#include "utils/cmdline.h"
#include "driverlib/uart.h"
#include "driverlib/interrupt.h"

#include "DelayTimer.h"
```

B. Enabling UART Module in TM4C123GH6PM

To enable UART module 1 on the Tiva C Series microcontroller, first, set bit 1 in the RCGCUART register to activate the clock for UART module 1. Since UART module 1 pins are on Port B (PB0 and PB1), enable the clock for GPIOB by setting bit 1 in the RCGCGPIO register. Next, configure the pins for digital functionality and alternate function by setting bits 0 and 1 in the GPIODEN register and bits 0 and 1 in the GPIOAFSEL register for Port B. To select UART as the alternate function, set bits 0 and 4 in the GPIOPCTL register for pins 0 and 1. This process ensures proper clocking and pin configuration for UART module 1, allowing seamless communication with external devices on PB0 and PB1.

C. Configuring UART protocol parameters

To enable UART module 1 on the Tiva C Series microcontroller, first, set bit 1 in the RCGCUART register to activate the clock for UART module 1. Since UART module 1 pins are on Port B (PB0 and PB1), enable the clock for GPIOB by setting bit 1 in the RCGCGPIO register. Next, configure the pins for digital functionality and alternate function by setting bits 0 and 1 in the GPIODEN register and bits 0 and 1 in the GPIOAFSEL register for Port B. To select UART as the alternate function, set bits 0 and 4 in the GPIOPCTL register for pins 0 and 1. This process ensures proper clocking and pin configuration for UART module 1, allowing seamless communication with external devices on PB0 and PB1.

D. Initializing GNSS:

The GNSSPWR function is responsible for enabling the GNSS power on the SIM808 module. It sends the command "AT+CGNSPWR=1" to turn on the GNSS power. The response is then checked using recvFind function within a timeout of 5000 milliseconds. If "OK" is received, the function returns true, indicating successful GNSS power activation. The GNSSEQ function configures the GNSS to report specific NMEA sentences, particularly the Recommended Minimum Navigation Information (RMC) sentence. It sends the command "AT+CGNSSEQ="RMC"" to set the GNSS to report RMC sentences. The response is checked using recvFind within a timeout of 5000 milliseconds. If "OK" is received, the function returns true, indicating successful configuration.

E. Retrieving GNSS Information:

This function retrieves GNSS information by sending the command "AT+CGNSINF" to the SIM808 module.
The response is checked using recvFind within a timeout of 5000 milliseconds. If "OK" is received, the function extracts and stores latitude, date-time, and longitude from the response into respective arrays (Latitude, DateTime, Longitude). The function returns true if the GNSS information retrieval is successful.

F. Configuring GPRS:

The GPRS configuration functions collectively prepare the SIM808 module for GPRS communication.

GPRS1(): Configures the GPRS context type by sending the command "AT+SAPBR=3,1,"Contype","GPRS". The command is instructing the SIM808 module to configure the GPRS bearer with a context ID of 3, set the parameter tag to 1, and specify the connection type as "GPRS".

.

GPRS2(): Sets the GPRS Access Point Name (APN) by sending the command "AT+SAPBR=3,1,\"APN\",\"airtelgprs.com\"". The command instructs the SIM808 module to configure the GPRS bearer with a context ID of 3, set the parameter tag to 1, and specify the Access Point Name (APN) as "airtelgprs.com." This command is an essential step in configuring the GPRS settings to establish a connection to the specified GPRS network (in this case, the "airtelgprs.com" network).

GPRS3(): Opens the GPRS bearer by sending the command "AT+SAPBR=1,1". The command instructs the SIM808 module to open the GPRS bearer. This command is a crucial step in establishing a GPRS connection, allowing the module to communicate over a GPRS network. After successfully executing this command, the module is ready to send and receive data over the GPRS network, enabling functionalities such as data transmission or accessing internet services.

G. Initializing HTTP service on SIM808 module:

The following functions are used to initialize HTTP communication on the SIM808 module:

HTTP1(): Initializes the HTTP service on the SIM808 module by sending the command "AT+HTTPINIT." Waits for a response using recvFind within a timeout of 5000 milliseconds. Returns true if "OK" is received, indicating successful initialization of the HTTP service.

HTTP2(): Sets the HTTP context identifier (CID) by sending the command "AT+HTTPPARA="CID",1.". The command "AT+HTTPPARA="CID",1" instructs the SIM808 module to set the HTTP context identifier (CID) to 1. The CID is used to specify the HTTP context under which subsequent HTTP commands will operate. By setting the CID to 1, the module ensures that it uses the appropriate context for subsequent HTTP actions, such as sending an HTTP GET request or other HTTP-related operations. It's a way of associating the HTTP communication with a specific context, allowing for multiple simultaneous HTTP sessions if needed.

H. Performing HTTP GET Requests:

The HTTPGet function is a crucial component of a program designed to facilitate communication with the SIM808 module for HTTP interactions. It orchestrates the process of sending an HTTP GET request to a specified server, incorporating latitude and longitude parameters into the request URL. The operation involves a series of well-defined steps:

Adjusting Latitude and Longitude:
The function first checks whether the latitude and longitude strings start with a comma. If they do, it adjusts the pointers to skip the leading comma.

Building the URL:
The function then constructs the URL for the HTTP GET request using the latitude and longitude values. This URL is formatted as "http://<server IP> /api/gps?latitude=%s&longitude=%s". The command to set this URL as a parameter is sent to the SIM808 module, and the function waits for an "OK" response within a specified timeout.

Performing HTTP Action:
Following successful URL parameter setting, the function initiates the HTTP action (GET request) by sending the appropriate AT command to the module. Again, it waits for an "OK" response within a predefined timeout.

Reading HTTP Response:
Subsequently, the function reads the HTTP response from the server by issuing the relevant AT command. It waits for and verifies the reception of an "OK" response within the stipulated timeout.

Returning Result:
Finally, the function aggregates the outcomes of these steps and returns a boolean result. The return value is true if all steps were successful, meaning that the HTTP GET request was effectively executed, and the server response was successfully read.

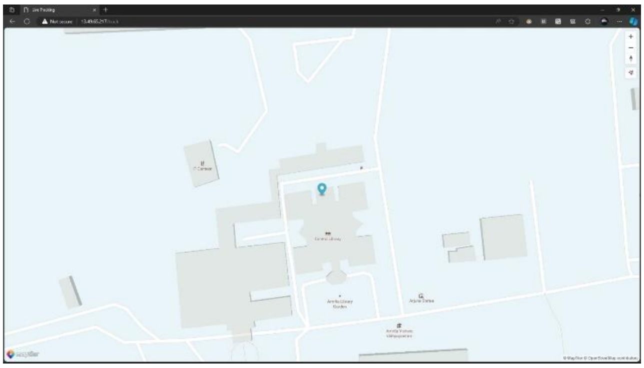

Figure4. Map UI Top View

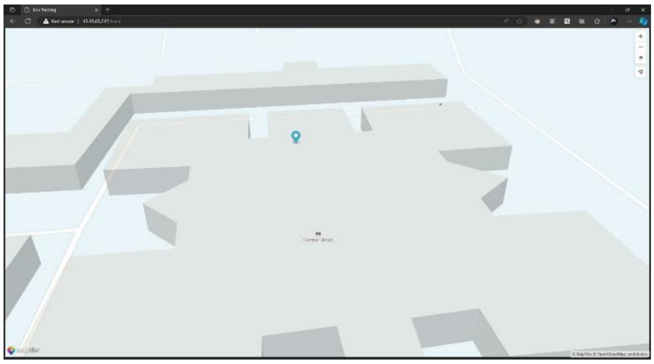

Figure5. Map UI 3D View

V. CONCLUSION AND FUTURE WORK

This project successfully implements a comprehensive communication system utilizing the SIM808 module, integrating GPS and GPRS capabilities. The code orchestrates the initialization of key functionalities, such as GPS positioning, GPRS configuration, and HTTP communication, through a series of AT commands. The modular design enhances readability and maintainability, allowing for easy integration of additional features. An intuitive and visually appealing User Interface (UI) was also developed using MapTiler and AWS services. The project, which involves real-time location tracking using SIM808 modules, AWS, and MapTiler, exhibits considerable potential for future enhancements. To fortify its capabilities, the project could benefit from the integration of advanced geospatial analytics for predictive insights and anomaly detection. Implementing geofencing and alert mechanisms would contribute to a more secure and responsive system. Mobile application development for iOS and Android platforms would enhance accessibility. Additionally, optimizing energy efficiency, ensuring regulatory compliance, and fostering integration with the broader IoT ecosystem are critical considerations for the project's evolution. Customizable user interfaces, community-sharing features, and offline data storage capabilities would contribute to a more versatile and user-friendly system. The project's future scope extends to embracing emerging technologies and staying adaptable to regulatory landscapes, ensuring its relevance and effectiveness in diverse scenarios.

REFERENCES

[1] Waleed Nassar. SMS example using Tiva C LaunchPad and SIM 808 GSM Module[Online]. Available: http://mcuhq.com/34/sms-example-using-tiva-c-launchpad-and-sim-808-gsm-module

[2] Texas Instruments, "Tiva™ TM4C123GH6PM Microcontroller" TM4C123GH6PM datasheet, June. 2014

[3] Selcuk Cakmak. Tiva LaunchPad: Internet of Things (IoT) with ESP8266 WiFi[Online]. Available: https://www.hackster.io/selcuk-cakmak/tiva-launchpad-internet-of-things-iot-with-esp8266-wifi-3e00e2

[4] Yuval Adam. Tiva-c[Online]. Available https://github.com/yuvadm/tiva-c

[5] Microcontntrollers Lab. UART Communication TM4C123 Tiva C LaunchPad with Example Codes[Online]. Available: https://microcontrollerslab.com/uart-communication-tm4c123-tiva-c-launchpad/