L
A
M
P
I
R
A
N

Lampiran Listing Program

```
// TTGO T-Call pin definitions #define MODEM_RST        5

#define ESP32_PWKEY        4

#define ESP32_POWER_ON 23

#define ESP32_TX               27

#define ESP32_RX               26

#define I2C_SDA        21

#define I2C_SCL        22

#include <TinyGPS++.h>  //https://github.com/mikalhart/TinyGPSPlus  #include <AceButton.h> // https://github.com/bxparks/AceButton

#define BLYNK_PRINT Serial #define BLYNK_HEARTBEAT 30

#define TINY_GSM_MODEM_SIM800

#include <TinyGsmClient.h>  //  https://github.com/vshymanskyy/TinyGSM #include <BlynkSimpleSIM800.h> //https://github.com/blynkkk/blynk-library

#include <Wire.h>

// #include <TinyGsmClient.h> #include "utilities.h"

using namespace ace_button;//Buttons

#define SMS_Button 34

#define Call_Button 35


// Emergency Number and Message

String message = "It's an Emergency. I'm at this location "; String mobile_number = "Mobile Number with country code";

String message_with_data;

// Variables for storing GPS Data float latitude;

float longitude; float speed; float satellites;

String direction;

// Switch ButtonConfig config1;
```

AceButton call_button(&config1); ButtonConfig config2;

AceButton sms_button(&config2);

void handleEvent_call(AceButton*, uint8_t, uint8_t); void handleEvent_sms(AceButton*, uint8_t, uint8_t);// Set serial for GPS Module #define SerialMon Serial// Hardware Serial for builtin GSM Module #define SerialAT Serial1const char apn[] = "indosatgprs";const char user[] = "indosat"; const char pass[] = "indosat";// You should get Auth Token in the Blynk App.// Go to the Project Settings (nut icon).const char auth[] = "NwNPmXfICoPjklEnEpv9vrgF57GgTgA5";//static const int RXPin = 4, TXPin = 5; static const uint32_t GPSBaud = 9600;

TinyGPSPlus gps; WidgetMap myMap(V0);

//SoftwareSerial ss(RXPin, TXPin); BlynkTimer timer;TinyGsm modem(SerialAT); unsigned int move_index = 1;void setup(){// Set console baud rate

Serial.begin(9600); delay(10);// Keep power when running from battery Wire.begin(I2C_SDA, I2C_SCL);bool isOk = setPowerBoostKeepOn(1); SerialMon.println(String("IP5306 KeepOn ") + (isOk ? "OK" : "FAIL"));// Set-up modem reset, enable, power pins pinMode(MODEM_PWKEY, OUTPUT); pinMode(MODEM_RST, OUTPUT); pinMode(MODEM_POWER_ON, OUTPUT);

pinMode(SMS_Button, INPUT); pinMode(Call_Button, INPUT);digitalWrite(MODEM_PWKEY, LOW); digitalWrite(MODEM_RST, HIGH); digitalWrite(MODEM_POWER_ON, HIGH);// Set GSM module baud rate and UART pins

SerialAT.begin(115200, SERIAL_8N1, MODEM_RX, MODEM_TX);delay(3000);// Restart takes quite some time// To skip it, call init() instead of restart() SerialMon.println("Initializing modem..."); modem.restart();

String modemInfo = modem.getModemInfo(); SerialMon.print("Modem: "); SerialMon.println(modemInfo);

/ Unlock your SIM card with a PIN

//modem.simUnlock("1234");

SerialMon.print("Waiting for network..."); if (!modem.waitForNetwork(240000L)) { SerialMon.println(" fail");

delay(10000); return;}

```
 SerialMon.println(" OK");
if (modem.isNetworkConnected()) { SerialMon.println("Network connected");}
SerialMon.print(F("Connecting to APN: ")); SerialMon.print(apn);
if (!modem.gprsConnect(apn, user, pass)) { SerialMon.println(" fail");
delay(10000); return;}
SerialMon.println(" OK");
// ss.begin(GPSBaud);
Blynk.begin(auth, modem, apn, user, pass); timer.setInterval(5000L, checkGPS);
config1.setEventHandler(handleEvent_call);
config2.setEventHandler(handleEvent_sms);
call_button.init(Call_Button); sms_button.init(SMS_Button);}
void checkGPS(){
if (gps.charsProcessed() < 10){
//Serial.println(F("No GPS detected: check wiring.")); Blynk.virtualWrite(V4, "GPS ERROR");}}
void loop(){while (Serial.available() > 0){
if (gps.encode(Serial.read())) displayInfo();}
Blynk.run();
timer.run();sms_button.check(); call_button.check();}
void displayInfo(){
if (gps.location.isValid() ){
latitude = (gps.location.lat()); //Storing the Lat. and Lon. longitude = (gps.location.lng());
//Serial.print("LAT: ");
//Serial.println(latitude, 6); // float to x decimal places
//Serial.print("LONG: ");
//Serial.println(longitude, 6);
Blynk.virtualWrite(V1, String(latitude, 6));
```

```
Blynk.virtualWrite(V2, String(longitude, 6)); myMap.location(move_index, latitude, longitude, "GPS_Location"); speed = gps.speed.kmph(); //get speed Blynk.virtualWrite(V3, speed);

direction = TinyGPSPlus::cardinal(gps.course.value()); // get the direction Blynk.virtualWrite(V4, direction);

satellites = gps.satellites.value(); //get number of satellites Blynk.virtualWrite(V5, satellites);}//Serial.println();}

void handleEvent_sms(AceButton* /* button */, uint8_t eventType, uint8_t /* buttonState */) {

switch (eventType) {

case AceButton::kEventPressed:

// Serial.println("kEventPressed");

message_with_data = message + "Latitude = " + (String)latitude + "Longitude = " + (String)longitude;

modem.sendSMS(mobile_number, message_with_data); message_with_data = "";

break;

case AceButton::kEventReleased:

//Serial.println("kEventReleased"); break;}}

void handleEvent_call(AceButton* /* button */, uint8_t eventType, uint8_t /* buttonState */) {

switch (eventType) {

case AceButton::kEventPressed:

// Serial.println("kEventPressed"); modem.callNumber(mobile_number); break;

case AceButton::kEventReleased:

//Serial.println("kEventReleased"); break;}}
```