

```

/
*****
*****
* PROGRAMA ENVIA UM PACOTE E SALVA A POSIÇÃO DO GPS NO CARTÃO SD *
*****
*****/

#include <SoftwareSerial.h>
#include <TinyGPS.h>
#include <SdFat.h>
#include <SPI.h>
#include <LoRa.h>
SdFat sdCard;
SdFile meuArquivo;
SoftwareSerial serial1(5, 6); // RX, TX
TinyGPS gps1;
int counter1=0;
int pacote=0;
#define ledLoRa 8
#define ledSD 7
#define chave 2
const int chipSelect = 4;
void setup() {
  serial1.begin(9600);
  if (!LoRa.begin(915E6)) {
    while (1);
  }
  LoRa.setTxPower(14);
  LoRa.setSpreadingFactor(12);
  LoRa.setSignalBandwidth(125E3);
  //LoRa.setCodingRate4(8);
  LoRa.disableCrc();
  pinMode(ledSD,OUTPUT);
  digitalWrite(ledSD,LOW);
  pinMode(ledLoRa,OUTPUT);
  digitalWrite(ledLoRa,LOW);
  pinMode(chave,INPUT);

  //TESTE DO CARTÃO SD
  bool testesd= false;

  if(!sdCard.begin(chipSelect,SPI_HALF_SPEED))
  {
    sdCard.initErrorHalt();
    digitalWrite(ledSD, HIGH);
    delay(500);
    digitalWrite(ledSD, LOW);
    delay(500);
    digitalWrite(ledSD, HIGH);
    delay(500);
    digitalWrite(ledSD, LOW);
    testesd=true;
  }
  if (!meuArquivo.open("teste1.txt", O_RDWR | O_CREAT | O_AT_END))//MUDE AQUI O NOME DO
  //ARQUIVO AO QUAL VOCÊ ESTÁ USANDO OU VAI SALVAR
  {
    digitalWrite(ledSD, HIGH);

```

```

        delay(1000);
        testesd=false;
    }
    delay(1000);
    delay(1000);
    digitalWrite(ledSD, LOW);
    meuArquivo.println("Medidas de altitude e posição com GPS");
    while(digitalRead(chave) ==0)
    {
        digitalWrite(ledLoRa, HIGH);
        delay(500);
        digitalWrite(ledLoRa, LOW);
        delay(500);
        digitalWrite(ledSD, HIGH);
        delay(500);
        digitalWrite(ledSD, LOW);
    }

}

void loop() {
    //Serial.print("Sending packet: ");
    //Serial.println(counter);
    bool cardsd = true;
    bool recebido, teste = false;
    static unsigned long delayPrint;

    // send packet
    while (serial1.available()) {
        char cIn = serial1.read();
        recebido = (gps1.encode(cIn) || recebido); //Verifica até receber o primeiro sinal dos satélites
    }
    if ( (recebido) && ((millis() - delayPrint) > 1000) ) { //Mostra apenas após receber o primeiro sinal.
        Após o primeiro sinal, mostra a cada segundo.
        delayPrint = millis();
        float latitude, longitude; //As variáveis podem ser float, para não precisar fazer nenhum cálculo
        unsigned long idadeInfo;
        gps1.f_get_position(&latitude, &longitude, &idadeInfo); //O método f_get_position é mais
        indicado para retornar as coordenadas em variáveis float, para não precisar fazer nenhum cálculo

        int pause=0;
        if (latitude != TinyGPS::GPS_INVALID_F_ANGLE)
        {
            if(pause==0)
            {

                if (latitude != TinyGPS::GPS_INVALID_F_ANGLE) {
                    //Serial.print("Latitude: ");
                    //Serial.println(latitude, 6); //Mostra a latitude com a precisão de 6 dígitos decimais
                    meuArquivo.print("LAT ");
                    meuArquivo.print(latitude, 6);

                    if (longitude != TinyGPS::GPS_INVALID_F_ANGLE) {
                        //Serial.print("Longitude: ");
                        //Serial.println(longitude, 6); //Mostra a longitude com a precisão de 6 dígitos decimais

```

```

    meuArquivo.print(" LONG ");
    meuArquivo.println(longitude, 6);
    meuArquivo.print("Link para Google Maps: https://maps.google.com/maps/?z=10&q=");
    meuArquivo.print(latitude,6);
    meuArquivo.print(",");
    meuArquivo.println(longitude, 6);
}
meuArquivo.print("posição: ");
meuArquivo.println(counter1);
for(pacote=1;pacote<=10;pacote++)
{
    LoRa.beginPacket();
    LoRa.print(counter1);
    LoRa.endPacket();
    delay(1000);
    meuArquivo.print("pacote: ");
    meuArquivo.println(pacote);
}
}
}
if(digitalRead(chave)==0){
    meuArquivo.close();
while(digitalRead(chave)==0)
{
    digitalWrite(ledSD, HIGH);
    delay(500);
    digitalWrite(ledSD, LOW);
    delay(500);
}
}
pause=1;
while(digitalRead(chave)==1)
{

    digitalWrite(ledLoRa, HIGH);

    digitalWrite(ledSD, HIGH);
    while (serial1.available()) {
        char cln = serial1.read();
        recebido = (gps1.encode(cln) || recebido); //Verifica até receber o primeiro sinal dos satélites
    }
    if ( (recebido) && ((millis() - delayPrint) > 1000) ) { //Mostra apenas após receber o primeiro sinal.
//Após o primeiro sinal, mostra a cada segundo.
        delayPrint = millis();
        unsigned long idadeInfo;
        gps1.f_get_position(&latitude, &longitude, &idadeInfo); //O método f_get_position é mais
//indicado para retornar as coordenadas em variáveis float, para não precisar fazer nenhum cálculo
    }
    digitalWrite(ledSD, LOW);
    digitalWrite(ledLoRa, LOW);
}
    pause = 0;
    counter1++;
}
}

```

```
}  
  else  
  {  
    digitalWrite(ledSD, HIGH);  
    delay(150);  
    digitalWrite(ledSD, LOW);  
    delay(150);  
  }  
}
```