

```
// Inclusão das bibliotecas
Inclua <ESP8266Libs>
```

```
// Definição de pinagem do switch Remove Before Flight
Define RBF_PIN como GPIOX
Define I2C_MASTER como endereço I2C_MASTER
Define I2C_SLAVE como endereço I2C_SLAVE
```

```
Variável RBF_state: booleano //Variável de estado do switch remove before flight
Variável address: inteiro //Endereço de memória
```

```
Função setup(): void //Configurações iniciais
//inicializa serial
//define modo de pinagem do RBF_PIN
EEPROM_begin(EEPROM_SIZE) // Inicia EEPROM
address = 0 //Reinicia endereço de memória
RBF_state = leitura_estado(RBF_PIN) //Lê o estado do switch rbf
Fim função
```

```
Função loop(): void //Rotina principal
se RBF_state:
    setup_provide_data()
senão:
    setup_flight()
Fim função
```

```
Função setup_provide_data(): void
IniciaWebServer(80) //Inicia o web server com protocolo HTTP
WiFi(modo_acess_point) //Seta o módulo wifi como ponto de acesso
WiFi_start(ssid, password); // Inicia o wifi com nome de rede e senha
Enquanto (WifiStatus != Conectado) loop //Aguarda conexao
    espera(tempo)
fim loop
server_on("/", handle_OnConnect) // Configura Handles do Server e Inicia Server
server_onNotFound(handle_NotFound)
server_start()
provide_flight_data() //Loop para envio de dados
Fim função
```

```
Função setup_flight(): void
Conexao(Pino_SDA, Pino_SCL, I2C_MASTER) //Cria conexao com o BMP280 por
interface I2C
    read_flight_data() //Loop para leitura dos dados
Fim função
```

```
Função provide_flight_data(): void
Enquanto RBF_state = True loop:
    server_handleClient() // Faz o Handle
```

```
RBF_state = leitura_estado(RBF_PIN)
Fim loop
```

```
Função read_flight_data(): void
    interrupcao = polling(perodo) //Define a interrupção por polling periódica com período igual
    a 20ms
    variavel periodos = 0: inteiro
    variavel amostras[5]: inteiro //Lista para armazenar as amostras
    Enquanto RBF_state = False loop:
        se(interrupcao):
            dado = request_from(I2C_SLAVE) //Lê dado do BMP280
            amostras[periodos] = dado
            periodos = periodos + 1
        se(periodos = 4):
            media = media_movel(periodos) //Faz a média movel dos periodos, armazenando um
            dado a cada 100ms
            periodos = 0
            EEPROM_write(address, media) //Escreve na EEPROM
            address += sizeof(media) //Incrementa memória
            RBF_state = leitura_estado(RBF_PIN)
    Fim loop
Fim função
```

```
Função handle_OnConnect(): void //Conexao com o servidor
    server_send(200, "text/html", GetDados) //Lê dados da memória EEPROM e retorna para o
    usuário
    //A partir dos dados, computa o tempo e altura de apogeu e envia para o usuário
Fim função
```

```
Funcao handle_NotFound(): void //Falha de conexao com o servidor
    server_send(404, "text/plain", "Not found")
Fim função
```