

```
// >--> MODULO SENSOR UMIDADE SOLO
// >--> TEMPO ENTRE MEDIDAS CONSECUTIVAS 1 MINUTO! PARA TESTES EXCLUSIVAMENTE!!!
// >--> TEMPO ENTRE MEDIDAS CONSECUTIVAS 5 MINUTOS PARA PRODUCAO!!!
/* ADS1115 - This code is designed to work with the ADS1115_I2CADC I2C Mini
Module available from ControlEverything.com.
https://www.controleverything.com/content/Analog-Digital-Converters?
sku=ADS1115_I2CADC#tabs-0-product_tabset-2
https://github.com/ControlEverythingCommunity/ADS1115
/* - - - - -
/* MPX5700 - Series - freescale
* https://www.nxp.com/docs/en/data-sheet/MPX5700.pdf
0 to 700 kPa (0 to 101.5 psi) >-----> 0,00 to 7 bar
15 to 700 kPa (2.18 to 101.5 psi) >--> 0,15 to 7 bar
0.2 to 4.7 V Output >--> 700 kPa = 4,7 V >--> 1013 hPa >--> 101,3 cBar
Logo para a pressao atmosferica de 1013 hPa >--> Vout = 0,680157142 V
Para 0,8465 V que esta medindo agora é equivalente a Patm = 126,07 kPa. */
/* - - - - -
/* 1 centiBar = 1 kPa = 10 hPa >--> 50 centibar = 50 kPa >--> valor
tipico de presao para medida de tensao so solo! */
/* - - - - -
// VERSAO PARA TESTES - NAO COMPATIVEL COM SISTEMA DO OLIVO
// >--> TEMPO ENTRE MEDIDAS CONSECUTIVAS 1 MINUTO!
// intervalo = 60000; // TEMPO DE SUBIDA PARA TESTES DE SENSORES BME280
// VERSAO 2 - TAVARES - LED1 MUDOU PARA ATL-5 ANTES ERA LED0 NO ATL-3
// VERSAO PARA TESTES NO SISTEMA ANTIGO DO RPi - LINUX/APACHE/MYSQL/PHP
/* https://github.com/esp8266/Arduino
Arduino core for ESP8266 WiFi chip This project brings support for
ESP8266 chip to the Arduino environment. It lets you write sketches using
familiar Arduino functions and libraries, and run them directly on
ESP8266, no external microcontroller required. ESP8266 Arduino core comes
with libraries to communicate over WiFi using TCP and UDP, set up HTTP,
mDNS, SSDP, and DNS servers, do OTA updates, use a file system in flash
memory, work with SD cards, servos, SPI and I2C peripherals. */
/* - - - - -
/*
VERSÃO RPi TESTE DO CODIGO - UPDATED 050118
VERAO 3.0 DATA: 01072017
COMPILADO NA VERSAO ARDUINO: 1.8.1

PLACA WIFI ESP8266-07 AT THINKER
PROGRAMA: MINI ESTACAO CLIMATICA
CONTÃ%oM SENSORES: BMP-180 E DHT22

CONFIGURACAO PLACA GRAVACAO - ESP-07
FUNCIONA COM BIBLIOTECE ESP-07 COMMUNITY ATE VERSAO 2.3.0
ATENCAO - VERSAO 2.4 NAO FUNCIONA PARA ESTE MODELO ESP-07

ATENCAO NAO COMPILAR ESP-07 NA VERSAO 2.4 OU SUPERIOR!!!!

PLACA: GENERIC ESP8266 MODULE
FLASH MODE: DIO
FLASH SIZE: 1M (512K SPIFFS)
DEBUG PORT: DISABLED <--<
DEBUG LEVEL: RIEN <--<
RESET MOTHOD: ck
FLASH FREQUENCY: 40 MHz
```

CPU FREQUENCY: 80 MHz
UPLOAD SPEED: 115200
PORTA: PORTA ESP CONECTADA AO COMPUTADOR

CONFIGURACAO PLACA GRAVACAO - ESP-12E

PLACA: NODE MCU 1.0 (ESP-12E MODULE)
VERSAO LIB ESP 2.4.1 --> 2.4.2 NAO FUNC 30/09/2018
CPU FREQUENCY: 80 MHz
FLASH SIZE: 4M (1M SPIFFS)
UPLOAD SPEED: 115200
PORTA: PORTA ESP CONECTADA AO COMPUTADOR

```
*/
/* - - - - -
// AGROTECHLINK.COM - ESP8266 - PROGRAM HEADER TEMPLATE - 2017 - NOVEMBER
//
//          TODOS OS DIREITOS SAO RESERVADOS
/* - - - - -
/*  CADA GPIO POSSUI UMA IDENTIFICACAO ESPECIFICA
    PORTAS UTILIZADAS NAS PLACAS DA MINI ESTACAO CLIMATICA
    ATL3      >--> GPIO-16
    ATL4      >--> GPIO-14
    ATL5      >--> GPIO-12 + LED1
    ATL7      >--> GPIO-05 + SCL >--> PULLUP INTERNO / SENSOR BMP-180 (PRESSAO)
    ATL8      >--> GPIO-04 + SDA >--> PULLUP INTERNO / SENSOR BMP-180 (PRESSAO)
    /RST      >--> ---[10k]--+3V3  -//- JUMPER COM 0V P/ "RESET"
                +++[103]---CERAMIC CAPACITOR
    CH-PD     >--> ---[10k]--+3V3 + 103 CERAMIC CAPACITOR
                +++[103]---CERAMIC CAPACITOR
    GPIO-02   >--> ---[10k]--+3V3
    GPIO-00   >--> ---[10k]--+3V3  -//- JUMPER COM 0V P/ "FLASH"
    GPIO-15   >--> ---[10k]---0V
    RX + TX   >--> CONEXOES PARA GRAVADOR EXTERNO
    PROCEDIMENTO PARA GRAVACAO COM GRAVADOR FTDI
    FTDI-TX   >--> ATL-RX
    FTDI-RX   >--> ATL-TX
    FTDI-3V3   >--> ATL-3V3
    FTDI-0V    >--> ATL-0V
    NUNCA ALIMENTAR ESTE MODULO DIRETAMENTE PELO GRAVADOR
    OU USB DO COMPUTADOR! */
/* - - - - -
// LIVRARIAS EXTERNAS PARA FUNCIONAMENTO DOS SENSORES - CONEXAO - DADOS
/* - - - - -
#include <ESP8266WiFi.h>          // BIBLIOTECA WiFi DO ESP8266
#include <Wire.h>                  // NECESSÁRIO PARA COMUNICACAO I2C (PRESSAO)
#include "ADS1115.h"              // https://github.
com/ControlEverythingCommunity/ADS1115/tree/master/Arduino
#include <MySQL_Connection.h>      // CONEXAO COM BANCO DE DADOS
#include <MySQL_Cursor.h>          // CONEXAO COM BANCO DE DADOS
/* - - - - -
// AGROTECHLINK MINI ESTACAO CLIMATICA - PINOUTS - DEFINES - DESCRICOES
/* - - - - -
#define ATL3      16              // GPIO-16 | + LED0 >--> PRIMEIRA PLACA
#define ATL4      15              // GPIO-15 + ESTADO NORMAL DO ESP / PERMITE
ROTINAS E RESTART
#define ATL5      12              // GPIO-12 + LED1 >--> PLACAS VERSAO TAVARES
#define ATL7      5               // GPIO-05 + SCL >--> PULLUP INTERNO / SENSOR
```

```

BMP-180 (PRESSAO)
#define          ATL8              4              // GPIO-04 + SDA >--> PULLUP INTERNO / SENSOR
BMP-180 (PRESSAO)
#define          ATL9              2              // GPIO-02 + LED NATIVO DO ESP8266 / PERMITE
ROTINAS E RESTART
/* - - - - - */
// SENSOR PINS E CONFIGURACAO Wi-Fi
/* - - - - - */
#define          WIFI_SSID          "agrotechlink"    // NOME DA INTERNET DO RASPBERRY-PI
#define          WIFI_PASSWORD      "agricultura"    // SENHA DA INTERNET
/* - - - - - */
// DEFINICAO DAS VARIAVEIS GLOBAIS
/* - - - - - */
char             MAC[25];                // VARIAVEL MAC PARA O MySQL
String           mac;                    // VARIAVEL MAC STRING TO CHAR PARA O MySQL
int16_t          V_mpx;                  // SENSOR UMIDADE DO SOLO
unsigned long     tempoPrevio = 0;        // VARIAVEL DE CONTROLE DE TEMPO
unsigned long     intervalo = 20000;      // VARIAVEL PARA CONTROLE DE SUBIDA DOS
DADOS (1.ª SUBIDA = 45 SEGUNDOS)
ADS1115          ads;
char             INSERT_SQL[] = "INSERT INTO agrotech_intel.dia_clima SET mac='%s',
mpx_S=%s, hora=CURRENT_TIME, dia=CURRENT_DATE";
/* - - - - - */
// CONFIGURACOES DE ACESSO AO BANCO DE DADOS E Wi-Fi
/* - - - - - */
IPAddress        server_addr (192, 168, 42, 1);    // IP REDE WIFI GATEWAY RPi
char             user[] = "agrotech_u_intel";      // USUARIO DO BANCO DE DADOS
char             password[] = "OlvAgrotechlink1357"; // SENHA DO USUARIO
WiFiClient client;
MySQL_Connection conn((Client *)&client);
/* - - - - - */
// SENSOR DE UMIDADE DO SOLO >---> FRESCALE + ADS1151 + I2C
/* - - - - - */
// SETTING ADC PGA 4 CH >---> PARAMETERS FUNCTION
/* - - - - - */
void ADSconfig(){
// The ADC gain (PGA), Device operating mode, Data rate
// can be changed via the following functions
/* - - - - - */
    // ads.setGain(GAIN_TWO);          // 2x gain    +/- 2.048V  1 bit = 0.0625mV (default)
    // ads.setGain(GAIN_TWOTHIRDS);    // 2/3x gain +/- 6.144V  1 bit = 0.1875mV
ads.setGain(GAIN_ONE);                // 1x gain    +/- 4.096V  1 bit = 0.125mV
    // ads.setGain(GAIN_FOUR);         // 4x gain    +/- 1.024V  1 bit = 0.03125mV
    // ads.setGain(GAIN_EIGHT);        // 8x gain    +/- 0.512V  1 bit = 0.015625mV
    // ads.setGain(GAIN_SIXTEEN);      // 16x gain   +/- 0.256V  1 bit = 0.0078125mV
/* - - - - - */
ads.setMode(MODE_CONTIN);             // Continuous conversion mode
    // ads.setMode(MODE_SINGLE);       // Power-down single-shot mode (default)
/* - - - - - */
ads.setRate(RATE_128);                // 128SPS (default)
    // ads.setRate(RATE_8);            // 8SPS
    // ads.setRate(RATE_16);           // 16SPS
    // ads.setRate(RATE_32);           // 32SPS
    // ads.setRate(RATE_64);           // 64SPS
    // ads.setRate(RATE_250);          // 250SPS

```

```

        // ads.setRate(RATE_475);          // 475SPS
        // ads.setRate(RATE_860);          // 860SPS
/* - - - - - */
        ads.setOSMode(OSMODE_SINGLE);    // Set to start a single-conversion
/* - - - - - */
        ads.begin();}
/* - - - - - */
void setup() {
    pinMode(ATL5, OUTPUT);        digitalWrite(ATL5, HIGH);    // GPIO-16 + LED0 / INICIA
HIGH E TERMINA SETUP LOW
    pinMode(ATL4, OUTPUT);        digitalWrite(ATL4, HIGH);    // GPIO-15 + ESTADO NORMAL DO
ESP / HIGH
    pinMode(ATL9, OUTPUT);        digitalWrite(ATL9, HIGH);    // GPIO-02 + ESTADO NORMAL DO
ESP / HIGH
/* - - - - - */
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    while (WiFi.status() != WL_CONNECTED) {
        yield();}
/* - - - - - */
    mac = WiFi.macAddress();
/* - - - - - */
    // unsigned num = 1;
    unsigned mysqlResposta;
    mysqlResposta = conn.connect(server_addr, 3306, user, password);
    while (conn.connect(server_addr, 3306, user, password) != true) {yield();}
/* - - - - - */
    digitalWrite(ATL5, LOW);    // GPIO-16 + LED0 / DESLIGA SETUP OK!
    ADSconfig();}
/* - - - - - */
    // FIM DO SETUP E CONFIGURACOES. INICIO DO LOOP.
/* - - - - - */
void loop() {
    unsigned long currentMillis = millis();
    if (currentMillis - tempoPrevio >= intervalo) {        // SOBE OS PRIMEIROS DADOS NO
PRIMEIRO MINUTO
        digitalWrite(ATL5, HIGH);                            // GPIO-16 + LED0 | LIGADO. ESTOU
VIVO!
        tempoPrevio = currentMillis;
//        intervalo = 300000;                                // 5 MINUTOS (TEMPO DE SUBIDA)
        intervalo = 60000;                                    // 1 MINUTO (TEMPO DE SUBIDA PARA TESTES DE
SENSORES BME280)
/* - - - - - */
        V_mpx = ads.Measure_SingleEnded(0);
/* - - - - - */
// ATENCAO OLIVO --> V_mpx --> NO BANCO PASSA PARA INTEIRO E COMPRIMENTO 6
/* - - - - - */
        char SV_mpx[6], query[170];//, SC_cnt[255];
// CONVERTENDO DADOS DOS SENSORES PARA STRINGS
        dtostrf(V_mpx, 6, 6, SV_mpx);
        mac.toCharArray(MAC, 25);
/* - - - - - */
        sprintf(query, INSERT_SQL, MAC, SV_mpx);//, SC_cnt);
// CONCATENANDO A STRING INSERT_SQL PARA GRAVACAO NO BANCO DE DADOS
        MySQL_Cursor *cur_mem = new MySQL_Cursor(&conn);
        cur_mem->execute(query);    // SUBINDO DADOS PARA O BANCO

```

```

        delete cur_mem;                                // DELETANDO A QUERY EXECUTADA DA MEMORIA
    } yield();
    digitalWrite(ATL5, LOW);}    // LED1 | DESLIGA AO FINAL DO ENVIO PARA O RPi
/* - - - - - */
// MAIN FUNCTION END - FINAL
/* - - - - - */

```