**1/03/2024\_A**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

char inChar; //Guarda valor ingresado por teclado

int contador = 1; //Contador para controlar a que variable se va a ingresar dato

char ID1 = 'a'; //Variable para almacenar primer ID

char ID2 = 'b'; //Variable para almacenar segundo ID

char ID3 = 'c'; //Variable para almacenar tercer ID

char ID4 = 'd'; //Variable para almacenar cuarto ID

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Ingresar 'p' para asignar ID"); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(6000);

}

void loop(void)

{

if (Serial.available() > 0){ //Validación de datos recibidido en puerto serial, cara de surtidor a leer

delay(100);

X = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion se asignan los ID a cada puerto del arduino

{

if (Serial.available()) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

if (contador == 1)

{

ID1 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado ID1= ");

Serial.println(ID1);

contador = contador + 1;

}

else if (contador == 2)

{

ID2 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado ID2= ");

Serial.println(ID2);

contador = contador + 1;

}

else if (contador == 3)

{

ID3 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado ID3= ");

Serial.println(ID3);

contador = contador + 1;

}

else if (contador == 4)

{

ID4 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado ID4= ");

Serial.println(ID4);

contador = 1;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

}

if ((iB1.search(addr1) != 0) && (X==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud){

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.println();

delay(3000);

}

**1/03/2024\_B**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

int contador = 1; //Contador para controlar a que ID se va a ingresar dato

//Variables de ID´s

char ID1 = 'a';

char ID2 = 'b';

char ID3 = 'c';

char ID4 = 'd';

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0){ //Validación de datos recibidido en puerto serial, cara de surtidor a leer

delay(100);

X = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available()) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

if (contador == 1)

{

ID1 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

contador = contador + 1;

}

else if (contador == 2)

{

ID2 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

contador = contador + 1;

}

else if (contador == 3)

{

ID3 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

contador = contador + 1;

}

else if (contador == 4)

{

ID4 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 1;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

}

if ((iB1.search(addr1) != 0) && (X==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud); //Funcion para leer iButton

}

else if (X==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud){

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.println();

delay(3000);

}

**1/03/2024\_C**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

int contador = 1; //Contador para controlar a que ID se va a ingresar dato

//Variables de ID´s

char ID1 = 'a';

char ID2 = 'b';

char ID3 = 'c';

char ID4 = 'd';

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available()) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

if (contador == 1)

{

ID1 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

contador = contador + 1;

}

else if (contador == 2)

{

ID2 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

contador = contador + 1;

}

else if (contador == 3)

{

ID3 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

contador = contador + 1;

}

else if (contador == 4)

{

ID4 = inChar;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 1;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

}

if ((iB1.search(addr1) != 0) && (X==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

//Serial.print(ID1);

//Serial.print("-");

//Serial.println();

if ((iB1.search(addr1) != 0) && (X==ID1)) //Corrección lectura

{

}

}

else if (X==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

//Serial.print(ID2);

//Serial.print("-");

//Serial.println();

if ((iB2.search(addr1) != 0) && (X==ID2)) //Corrección lectura

{

}

}

else if (X==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

//Serial.print(ID3);

//Serial.print("-");

//Serial.println();

if ((iB3.search(addr1) != 0) && (X==ID3)) //Corrección lectura

{

}

}

else if (X==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

//Serial.print(ID4);

//Serial.print("-");

//Serial.println();

if ((iB4.search(addr1) != 0) && (X==ID4)) //Corrección lectura

{

}

}

else if (X==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

Serial.print(X);

Serial.print("-");

Serial.println();

delay(3000);

}

**1/03/2024\_A**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

int contador = 1; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 0) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

if (isdigit(inChar))

{

// Convertir el byte a un número entero

numero = inChar - '0'; // Resta el valor ASCII de '0'

}

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

contador = contador + 1;

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

contador = contador + 1;

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

contador = contador + 1;

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 1;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

}

if (isdigit(X))

{

// Convertir el byte a un número entero

X1 = X - '0'; // Resta el valor ASCII de '0'

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

Serial.print(X);

Serial.print("-");

Serial.println();

delay(3000);

}

**4/03/2024\_A**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

/\*

if (Serial.available() > 0) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

limpiarBufferSerial();

if (isdigit(inChar))

{

// Convertir el byte a un número entero

numero = inChar - '0'; // Resta el valor ASCII de '0'

}

contador = contador + 1;

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

}

\*/

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

limpiarBufferSerial();

}

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

//contador = contador + 1;

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

//contador = contador + 1;

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

//contador = contador + 1;

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

if (isdigit(X))

{

// Convertir el byte a un número entero

X1 = X - '0'; // Resta el valor ASCII de '0'

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

Serial.print(X);

Serial.print("-");

Serial.println();

delay(3000);

}

**4/03/2024\_B**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

/\*

if (Serial.available() > 0) {

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

limpiarBufferSerial();

if (isdigit(inChar))

{

// Convertir el byte a un número entero

numero = inChar - '0'; // Resta el valor ASCII de '0'

}

contador = contador + 1;

// Limpiar el buffer del puerto serial para eliminar cualquier dato adicional

limpiarBufferSerial();

}

\*/

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

limpiarBufferSerial();

}

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

//contador = contador + 1;

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

//contador = contador + 1;

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

//contador = contador + 1;

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

//delay(100);

//inChar = Serial.read(); // Lee el caracter recibido

//inChar\_2 = Serial.read();

//limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

//Serial.print(numero);

//Serial.print("-");

//Serial.println();

//delay(3000);

}

**4/03/2024\_C**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

limpiarBufferSerial();

}

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

//contador = contador + 1;

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

//contador = contador + 1;

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

//contador = contador + 1;

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

}

**4/03/2024\_D**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(8000);

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

limpiarBufferSerial();

}

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID1= ");

Serial.println(ID1);

//contador = contador + 1;

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID2= ");

Serial.println(ID2);

//contador = contador + 1;

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID3= ");

Serial.println(ID3);

//contador = contador + 1;

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para ID4= ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando ID iButton 1=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("iButton 1 no detectado.\n");

delay(2000);

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando ID iButton 2=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("iButton 2 no detectado.\n");

delay(2000);

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando ID iButton 3=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 3 no detectado.\n");

delay(2000);

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando ID iButton 4=");

adicion (addr1, Longitud, X); //Funcion para leer iButton

Serial.print(X1);

Serial.print("-");

Serial.println();

delay(3000);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("iButton 4 no detectado.\n");

delay(2000);

}

}

else

{

Serial.println("Puerto serial preparado para recibir datos...");//Stan By del dispositivo

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

}

**4/03/2024\_D**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(1500);

Serial.println("Dispositivo activo.");

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

if (X == 'l') //Visualiza el ID de cada puerto

{

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

}

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

limpiarBufferSerial();

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando M1 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("M1 no detectado.\n");

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando M2 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("M2 no detectado.\n");

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando M3 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M3 no detectado.\n");

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando M4 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M4 no detectado.\n");

}

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){ //Imprime inicio de trama dirrección iButton

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) { //Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

}

void end\_trama(char X){ //Imprime final de trama

Serial.print(X1);

Serial.print("-");

Serial.println();

}

**4/03/2024\_D**

#include <OneWire.h>

#include <SoftwareSerial.h>

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Variables de ID´s

int ID1 = 1;

int ID2 = 2;

int ID3 = 3;

int ID4 = 4;

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(1500);

Serial.println("Dispositivo activo.");

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

if (X == 'l') //Visualiza el ID de cada puerto

{

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

}

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

if (contador == 1)

{

ID1 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

}

else if (contador == 2)

{

ID2 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

}

else if (contador == 3)

{

ID3 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

}

else if (contador == 4)

{

ID4 = numero;//Actualiza ID con último valor ingresado por puerto serial

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

limpiarBufferSerial();

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando M1 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("M1 no detectado.\n");

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando M2 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("M2 no detectado.\n");

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando M3 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M3 no detectado.\n");

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando M4 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M4 no detectado.\n");

}

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){ //Imprime inicio de trama dirrección iButton

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) { //Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

}

void end\_trama(char X){ //Imprime final de trama

Serial.print(X1);

Serial.print("-");

Serial.println();

}

**4/03/2024\_D**

#include <OneWire.h>

#include <SoftwareSerial.h>

#include <EEPROM.h> //Librería para escribir leer EEROM

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

//#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Recuperacion de informacion desde EEPROM

int ID1 = EEPROM.read(1);

int ID2 = EEPROM.read(2);

int ID3 = EEPROM.read(3);

int ID4 = EEPROM.read(4);

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

//SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

Serial.begin(9600);

Serial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

Serial.println("Espere un momento, por favor...");

delay(1500);

Serial.println("Dispositivo activo.");

}

void loop(void)

{

if (Serial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = Serial.read();

inChar\_2 = Serial.read();

if (X == 'l') //Visualiza el ID de cada puerto

{

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

}

limpiarBufferSerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (Serial.available() > 1)

{

delay(100);

inChar = Serial.read(); // Lee el caracter recibido

inChar\_2 = Serial.read();

limpiarBufferSerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

if (contador == 1)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID1 = EEPROM.read(contador); //Lectura de EEPROM

Serial.print("Caracter ingresado para M1 = ");

Serial.println(ID1);

}

else if (contador == 2)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID2 = EEPROM.read(contador); //Lectura de EEPROM

Serial.print("Caracter ingresado para M2 = ");

Serial.println(ID2);

}

else if (contador == 3)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID3 = EEPROM.read(contador); //Lectura de EEPROM

Serial.print("Caracter ingresado para M3 = ");

Serial.println(ID3);

}

else if (contador == 4)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID4 = EEPROM.read(contador); //Lectura de EEPROM

Serial.print("Caracter ingresado para M4 = ");

Serial.println(ID4);

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

limpiarBufferSerial();

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

Serial.println("Enviando M1 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

Serial.print("M1 no detectado.\n");

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

Serial.println("Enviando M2 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

Serial.print("M2 no detectado.\n");

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

Serial.println("Enviando M3 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M3 no detectado.\n");

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

Serial.println("Enviando M4 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

Serial.print("M4 no detectado.\n");

}

}

}

void limpiarBufferSerial() { //Funcion para borrar dato adional del Buffer

while (Serial.available() > 0) {

char datoDescartado = Serial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){ //Imprime inicio de trama dirrección iButton

Serial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) { //Agrega 0X00

Serial.print("0");

}

Serial.print(addr1[0], HEX);

Serial.print(":");

}

void end\_trama(char X){ //Imprime final de trama

Serial.print(X1);

Serial.print("-");

Serial.println();

}

**4/03/2024\_E**

#include <OneWire.h>

#include <SoftwareSerial.h>

#include <EEPROM.h> //Librería para escribir leer EEROM

//Pin para habilitar el transceptor del SN75176 escritura en el bus AB

#define DE 5

//Pines lectura de chips DS1990 4 unidades

OneWire iB1(11); // Cara a corresponde a D11

OneWire iB2(12); // Cara b corresponde a D12

OneWire iB3(2); // Cara c corresponde a D2

OneWire iB4(3); // Cara d corresponde a D3

char X; //Variabe que deternima la cara de lectura surtidor

int X1 = 0; //Para lectura de iButton guarda el caracter igresado en numérico

char inChar; //Guarda valor ingresado por teclado parq configuración ID's

char inChar\_2; //Guarda segundo valor (decenas)

int contador = 0; //Contador para controlar a que ID se va a ingresar dato

int numero = 0; //Almacena el valor ingresado a numérico

//Varaibles para valor ingresado en decenas

int numero\_dec = 0; //Almacena valor en decenas

int numero\_und = 0; //Almacena valor en unidades

//Recuperacion de informacion desde EEPROM

int ID1 = EEPROM.read(1);

int ID2 = EEPROM.read(2);

int ID3 = EEPROM.read(3);

int ID4 = EEPROM.read(4);

byte addr1[] = {1 , 2, 3, 4, 5, 6, 7, 8}; //Array para guardar direccion de iButton

byte Longitud = 7; //Longitud array que contiene el numero de caracterres iButton

SoftwareSerial mySerial(4, 6); // RX, TX

void setup(void) {

mySerial.begin(9600);

pinMode(DE, OUTPUT);

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.println("Si desea configurar ID's, ingresar 'p' para programar."); //Inicio programa

mySerial.println("Espere un momento, por favor...");

delay(1500);

mySerial.println("Dispositivo activo.");

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

void loop(void)

{

if (mySerial.available() > 0)//Validación de datos recibidido en puerto serial, cara de surtidor a leer

{

delay(100);

X = mySerial.read();

inChar\_2 = mySerial.read();

limpiarBuffermySerial();

while (X=='p') //Programacion de aplicacion, se asignan los ID a cada puerto del arduino

{

if (mySerial.available() > 1)

{

delay(100);

inChar = mySerial.read(); // Lee el caracter recibido

inChar\_2 = mySerial.read();

limpiarBuffermySerial();

if (isdigit(inChar) && isdigit(inChar\_2))

{

numero\_dec = inChar - '0';

numero\_und = inChar\_2 - '0';

numero = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

contador = contador + 1;

if (contador == 1)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID1 = EEPROM.read(contador); //Lectura de EEPROM

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("Caracter ingresado para M1 = ");

mySerial.println(ID1);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

else if (contador == 2)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID2 = EEPROM.read(contador); //Lectura de EEPROM

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("Caracter ingresado para M2 = ");

mySerial.println(ID2);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

else if (contador == 3)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID3 = EEPROM.read(contador); //Lectura de EEPROM

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("Caracter ingresado para M3 = ");

mySerial.println(ID3);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

else if (contador == 4)

{

EEPROM.write(contador, numero); //Escritura en EEPROM

ID4 = EEPROM.read(contador); //Lectura de EEPROM

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("Caracter ingresado para M4 = ");

mySerial.println(ID4);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

contador = 0;

X = 'o'; //Finaliza programacion de ID sale de while e ingresa a lectura de iButton

}

limpiarBuffermySerial();

}

}

if (isdigit(X) && isdigit(inChar\_2))

{

numero\_dec = X - '0';

numero\_und = inChar\_2 - '0';

X1 = (numero\_dec \* 10) + numero\_und; //Se calcula el valor de los dos digitos

}

if (X == 'l') //Visualiza el ID de cada puerto

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("Caracter ingresado para M1 = ");

mySerial.println(ID1);

mySerial.print("Caracter ingresado para M2 = ");

mySerial.println(ID2);

mySerial.print("Caracter ingresado para M3 = ");

mySerial.println(ID3);

mySerial.print("Caracter ingresado para M4 = ");

mySerial.println(ID4);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Lectura cara a

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.println("Enviando M1 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

if ((iB1.search(addr1) != 0) && (X1==ID1)) //Corrección lectura

{

}

}

else if (X1==ID1) //Falla de lectura por ausencia de dispositivo en cara a

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("M1 no detectado.\n");

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Lectura cara b

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.println("Enviando M2 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

if ((iB2.search(addr1) != 0) && (X1==ID2)) //Corrección lectura

{

}

}

else if (X1==ID2) //Falla de lectura por ausencia de dispositivo en cara b

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("M2 no detectado.\n");

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Lectura cara c

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.println("Enviando M3 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

if ((iB3.search(addr1) != 0) && (X1==ID3)) //Corrección lectura

{

}

}

else if (X1==ID3) //Falla de lectura por ausencia de dispositivo en cara c

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("M3 no detectado.\n");

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Lectura cara d

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.println("Enviando M4 =");

adicion (addr1, Longitud, X); //Funcion para leer iButton

end\_trama (X);

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

if ((iB4.search(addr1) != 0) && (X1==ID4)) //Corrección lectura

{

}

}

else if (X1==ID4) //Falla de lectura por ausencia de dispositivo en cara c

{

digitalWrite(DE, HIGH); //Se habilita para escritura el SN75176

mySerial.print("M4 no detectado.\n");

digitalWrite(DE, LOW); ////Se habilita para lectura el SN75176

}

}

}

void limpiarBuffermySerial() { //Funcion para borrar dato adional del Buffer

while (mySerial.available() > 0) {

char datoDescartado = mySerial.read(); // Leer y descartar el dato adicioanl

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*Funcion encargada de agregar un 0X00 a los números\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*inferiores a 0X10, para cumplimentar los 16 caracteres\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*del iButton e imprimir dato por puerto serial \*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void adicion(byte addr1[],byte Longitud, char X){ //Imprime inicio de trama dirrección iButton

mySerial.print("@");

for( Longitud = Longitud; Longitud > 0; Longitud--)

{

if (addr1[Longitud] < 0x10) {//Agrega 0X00

mySerial.print("0");

}

mySerial.print(addr1[Longitud], HEX);

}

if (addr1[Longitud] < 0x10) { //Agrega 0X00

mySerial.print("0");

}

mySerial.print(addr1[0], HEX);

mySerial.print(":");

}

void end\_trama(char X){ //Imprime final de trama

mySerial.print(X1);

mySerial.print("-");

mySerial.println();

}