**IDENTIFICACIÓN DE LA PRÁCTICA**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Tgo. en Desarrollo de Software** | | |  | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Sistemas Embebidos I** | | |  | **Clave:** | MPF3107DSO | **Revisión:** | **A** |
| **No. de Práctica:** | 6 | **Nombre de la práctica:** | | Mover un carácter | | | |  |
| **Profesor:** | Antonio Lozano González | | | | | | |  |
| **Alumno:** | Gisel Carpinteiro Aguirre | | | | | | **Registro:** | **16100716** |
| **Semestre:** | **7** | **Grupo:** | **A2** |  | **Período:** | **Febrero – Junio 2019** | **Fecha:** | 02-04-2019 |

# Objetivo

* Conocer las diversas herramientas para desarrollar un programa.
* Utilizar una placa programable para resolver distintos problemas.
* Utilizar un software y hardware libre y conocer las ventajas además de las desventajas de este.

# Descripción

Desarrollar un programa en lenguaje C++ con ayuda de la placa Arduino, el cual deberá de mover un carácter especial por la LCD utilizando al teclado.

# Código C++

#include<Keypad.h>

#include <LiquidCrystal.h>

LiquidCrystal lcd(28, 30, 32, 34, 36, 38);

const byte filas = 4;

const byte columnas = 4;

int real=0;

int lados=0;

byte pinesF[filas] = {53,51,49,47};

byte pinesC[columnas] = {45,43,41,39};

char tecla;

char teclas[filas][columnas] = {

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'\*','0','#','D'}

};

Keypad teclado = Keypad(makeKeymap(teclas), pinesF, pinesC, filas, columnas);

byte cora[] = {

B00000,

B00000,

B11011,

B11111,

B01110,

B00100,

B00000,

B00000

};

byte flechaarriba[] = {

B00000,

B00100,

B01110,

B11111,

B00100,

B00100,

B00000,

B00000

};

byte flechaabajo[] = {

B00000,

B00100,

B00100,

B11111,

B01110,

B00100,

B00000,

B00000

};

byte flechaderecha[] = {

B00000,

B00100,

B00110,

B11111,

B00110,

B00100,

B00000,

B00000

};

byte flechaizquierda[] = {

B00000,

B00100,

B01100,

B11111,

B01100,

B00100,

B00000,

B00000

};

void setup() {

Serial.begin(9600);

lcd.begin(20, 4);

lcd.createChar(0, cora);

lcd.createChar(1, flechaarriba);

lcd.createChar(2, flechaabajo);

lcd.createChar(3, flechaderecha);

lcd.createChar(4, flechaizquierda);

lcd.home();

lcd.setCursor(8, 0); //columna, filaa

lcd.write((byte) 1);

lcd.setCursor(9, 0); //columna, filaa

lcd.print(" = 6");

lcd.setCursor(8, 2); //columna, filaa

lcd.write((byte) 2);

lcd.setCursor(4, 2); //columna, filaa

lcd.print("B = ");

lcd.setCursor(10, 1); //columna, filaa

lcd.write((byte) 3);

lcd.setCursor(11, 1); //columna, filaa

lcd.print(" = A");

lcd.setCursor(6, 1); //columna, filaa

lcd.write((byte) 4);

lcd.setCursor(2, 1); //columna, filaa

lcd.print("C = ");

}

void loop() {

tecla = teclado.getKey();

if(tecla!=0){

switch(tecla){

case 66:

lcd.clear();

real= real+1;

if(real>=0){

lcd.setCursor(lados, real+1);

}

if(real>=4){

real=0;

}

lcd.setCursor(lados, real);

lcd.write((byte) 0);

break;

case 54:

lcd.clear();

real= real-1;

if(real>=0){

lcd.setCursor(lados, real-1);

}

lcd.setCursor(lados, real);

lcd.write((byte) 0);

if(real==-1){

real=3;

}

break;

case 65:

lados = lados+1;

if(lados>=0){

lcd.setCursor(lados-1, real);

lcd.clear();

}

if(lados>=20){

lados=0;

real = real;

}

lcd.setCursor(lados, real);

lcd.write((byte) 0);

break;

case 67:

lcd.clear();

lados= lados-1;

if(lados>=0){

lcd.setCursor(lados-1, real);

}

if(lados<=-1){

lados=19;

}

lcd.setCursor(lados, real);

lcd.write((byte) 0);

break;

}

}

}

# Observaciones

Para generar en esta práctica un carácter especial se utilizo un generador online el cual te da el código para colocarlo en el código, y solo se manda a llamar este carácter cada vez que se quiera utilizar con la función de lcd.write. Al hacer se esta práctica se me complico un poco debido a que el carácter se perdía cada vez que tenía que pasaba un borde, la solución fue el acomodo de los pasos dentro del programa.

# Conclusiones

El Arduino es una placa basada en un microcontrolador ATMEL además cuenta con una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores, el cual utilizado el lenguaje C++.