

TECNOLÓGICO DE ESTUDIOS SUPERIORES DE CHALCO

ING. SISTEMAS COMPUTACIONALES

PRACTICA CON SENSOR LM35

MATERIA: SISTEMAS PROGRAMABLES

PROFESOR: ALFREDO GALICIA MOYSEN

EQUIPO 3

AGUIRRE VELAZQUEZ LUIS RAYMUNDO

ESPINOZA SANCHEZ DANIEL ANTONIO

MEDINA GARCÍA JOSÉ

OLIVARES VARGAS LUIS ALBERTO

SORIANO LOPEZ ALBERTO

GRUPO:4701

CARRERA	PLAN DE ESTUDIO	CLAVE ASIGNATURA	NOMBRE DE LA ASIGNATURA
Ingeniería en Sistemas Computacionales			Sistemas Programables

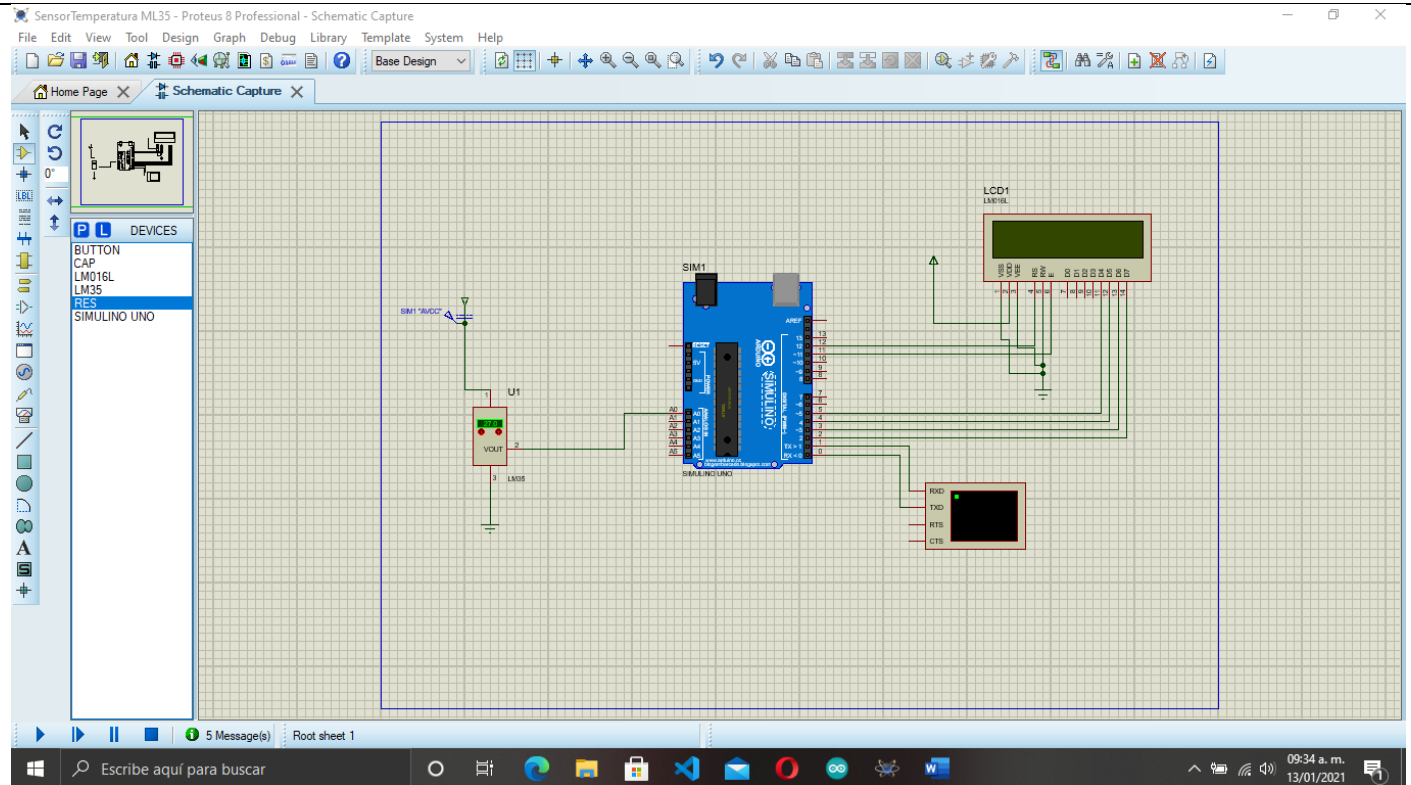
PRÁCTICA No.	LABORATORIO	SALÓN DE CLASE	DURACIÓN (HORA)
1	NOMBRE DE LA PRÁCTICA	SENSOR LM35	2 HRS

1	INTRODUCCIÓN
<p>Se nos ha solicitado realizar un ensamble de manera virtual de cómo funciona un sensor de temperatura con el dispositivo LM35 ya que nos permitirá conocer como funciona este tipo de componente para la medición de temperatura ya sea en un hospital así como también en un supermercado etc.</p>	

2	OBJETIVO (COMPETENCIA)
<ul style="list-style-type: none"> Implementar nuevas herramientas para tener más conocimiento acerca de cómo funcionan la programación de dispositivos en la actualidad. 	

3	CÓDIGO
	<pre>#include <LiquidCrystal.h> int analog_pin = A0; float tempC; float tempF; int tempdigital; LiquidCrystal lcd(12, 11, 5, 4, 3, 2); void setup(){ Serial.begin(9600); lcd.begin(16, 2); } void loop() { tempdigital= analogRead(analog_pin); tempC = (tempdigital * 5.0)*100.0/ 1024.0; tempF = tempC * 9.0 / 5.0 + 32.0; lcd.clear(); lcd.setCursor (0, 0); lcd.print ("Temp"); lcd.setCursor(6, 0); lcd.print(tempC); Serial.println("Temperatura"); Serial.println(tempC); delay(10); }</pre>
4	PROCEDIMIENTO (DESCRIPCIÓN)
<p style="text-align: center;">EQUIPO NECESARIO</p> <p>-Computadora Acer Aspire 5 con 12gb de ram y procesador I7 O I5</p> <p>-Software Arduino</p> <p>-Software Proteus</p>	<p style="text-align: center;">MATERIAL DE APOYO</p> <p>-Internet Explorer u otro navegador de su agrado</p>

Desarrollo de la Practica



Se implemento este código para el desarrollo y funcionamiento de nuestro ensamble del motor y sus velocidades que implementara y todo fue implementado para Arduino 1.

sensorTemplm35 Arduino 1.8.14 Hourly Build 2020/09/14 03:33

Archivo Editar Programa Herramientas Ayuda

```

sensorTemplm35
#include <LiquidCrystal.h>
int analog_pin = A0;
float tempC;
float tempF;
int tempdigital;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
}

void loop() {
  tempdigital= analogRead(analog_pin);
  tempC = (tempdigital * 5.0) * 100.0 / 1024.0;
  tempF = tempC * 9.0 / 5.0 + 32.0;

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Temp");
  lcd.setCursor(6, 0);
  lcd.print(tempC);
  Serial.println("Temperatura");
  Serial.println(tempC);
  delay(10);
}
  
```

Compilado

```

\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\core.a" "C:\\Users\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\WMath.cpp.o"
\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\core.a" "C:\\Users\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\WString.cpp.o"
\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\core.a" "C:\\Users\\Metal\\AppData\\Local\\Temp\\arduino_build_692638\\core\\abi.cpp.o"
  
```

1 - 26

Escribe aquí para buscar

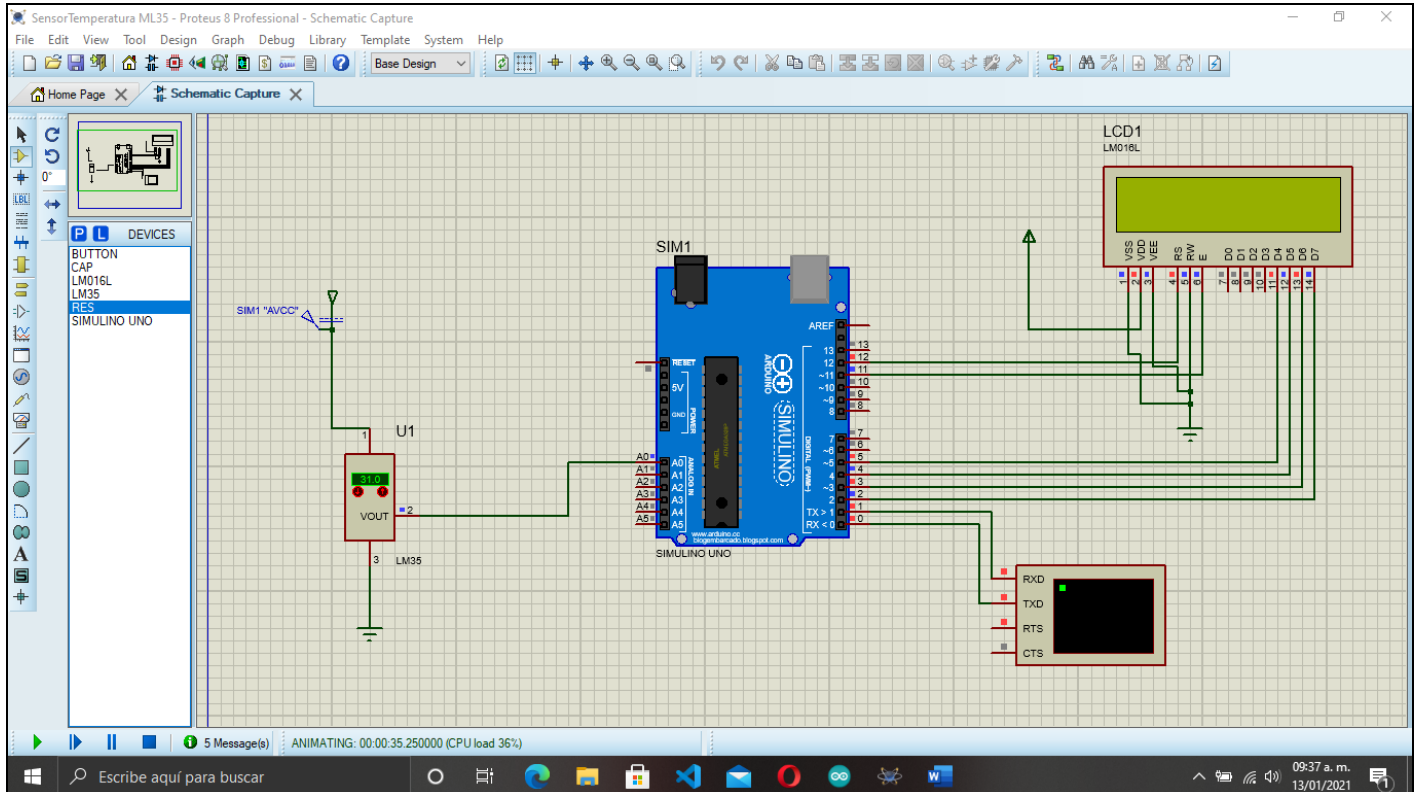
09:35 a. m.
13/01/2021

Aquí se muestra el código que desarrollamos para el funcionamiento del sensor LM35 para que funcione sin ningún problema.

C CÁLCULOS Y REPORTE

No fue necesario ingresar cálculos y solo necesitábamos saber los milisegundos equivalentes a segundos para implementar la medición

5 RESULTADOS Y CONCLUSIONES



El ensamblaje del circuito funciona perfectamente sin ningún problema.

```
sensorTemplm35 Arduino 1.8.14 Hourly Build 2020/09/14 03:33
Archivo Editar Programa Herramientas Ayuda

sensorTemplm35
#include <LiquidCrystal.h>
int analog_pin = A0;
float tempC;
float tempF;
int tempdigital;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
}

void loop() {
  tempdigital= analogRead(analog_pin);
  tempC = (tempdigital * 5.0) * 100.0 / 1024.0;
  tempF = tempC * 9.0 / 5.0 + 32.0;

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Temp");
  lcd.setCursor(6, 0);
  lcd.print(tempC);
  Serial.println("Temperatura");
  Serial.println(tempC);
  delay(10);
}

Compilado
El Sketch usa 4790 bytes (14%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 256 bytes (12%) de la memoria dinámica, dejando 1792 bytes para las variables locales. El máximo es 2048 bytes.
```

El código Arduino funciona perfectamente sin ningún problema alguno

Conclusión.

El objetivo de conocer los sensores ya que nos permitirán implementar nuevas estrategias y nuevas ideas para su uso y con el paso del tiempo poder hacer mejoras para su uso y sus estándares de calidad para que sean más accesibles a su venta al público.

También esto nos permitirá tener más conocimiento para el mundo laboral ya que esto nos da una idea de los productos que nos brindan y nos dan la idea de cómo se desarrolla una empresa para sacar su producto final.