Ministry of Education, Culture and Research of the Republic of Moldova

Technical University of Moldova

Department of Software and Automatic Engineering

REPORT

Laboratory Project nr.4
at Embeded Systems

Done by:
Gr.172-FAF Ion Dodon

Checked by: Andrei Bragarenco

Chișinău 2020

Laboratory Project Nr.4

Topic: Control

Objective: To control a lamp and a motor depending on a setpoint. To set the lamp on/off and to change the speed of the motor smoothly depending on the setpoint, the PID method and an encoder.

Domain:

A PID controller is an instrument used in industrial control applications to regulate temperature, flow, pressure, speed and other process variables. PID, which stands for proportional integral derivative, controllers use a control loop feedback mechanism to control process variables and are the most accurate and stable controller. In this ther article, how a PID works is explained in more detail.

PID control is a well-established way of driving a system towards a target position or level. It's a practically ubiquitous as a means of controlling temperature and finds application in myriad chemical and scientific processes as well as automation. PID control uses closed-loop control feedback to keep the actual output from a process as close to the target or setpoint output as possible.

Component description:

74HC595 - The 74HC595 consists of an 8−bit shift register and an 8−bit D−type latch with three−state parallel outputs. The shift register accepts serial data and provides a serial output. The shift register also provides parallel data to the 8−bit latch. The shift register and latch have independent clock inputs. This device also has an asynchronous reset for the shift register.

L298 MOTOR DRIVER - The L298 is an integrated monolithic circuit in a 15- lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

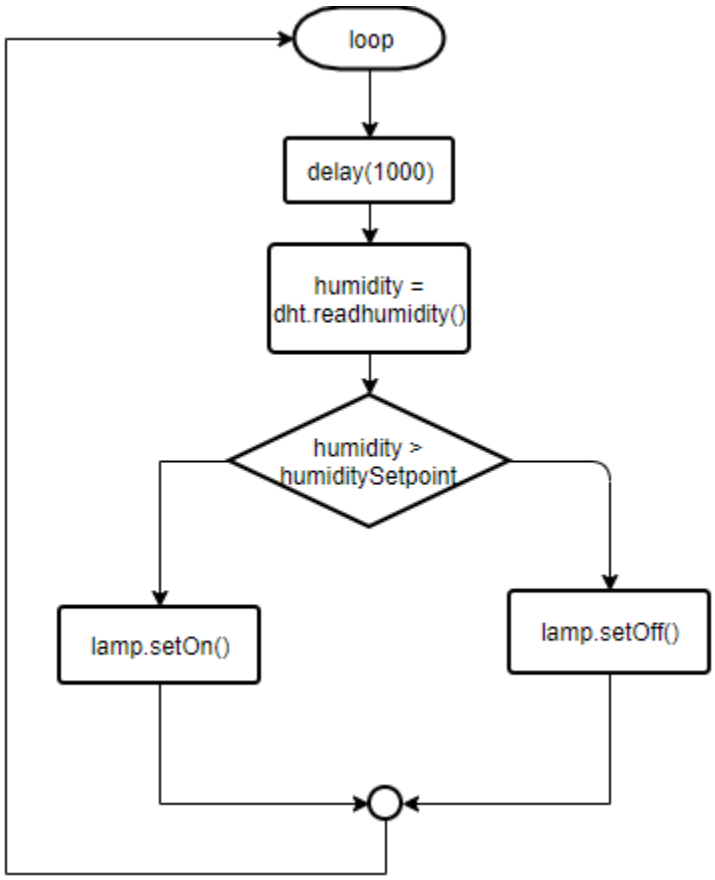
Motor - A **DC motor** is any of a class of rotary electrical **motors** that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields.

Electric lamp - An **electric lamp** is a conventional light emitting component used in different circuits, mainly for lighting and indicating purposes. The construction of lamp is quite simple, it has one filament surrounding which, a transparent glass made spherical cover is provided. The filament of the lamp is mainly made of tungsten as it has high melting point temperature. A lamp emits light energy as the thin small tungsten filament of lamp glows without being melted, while current flows through it.

Arduino Uno - The **Arduino Uno** is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.[2][3] The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.[4] It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo.[5][6] The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

Implementation:

a) **https://drive.google.com/open?id=1koWIEyEXZSGhabmaS1OpaWD9aBI7kNEY**



Conclusions: Working on this laboratory I have understood how to control a lamp (set on/off) dynamically in real time based on a setpoint give using the serial input. Also, I have got familiar with the different types of controlling a lamp, motor, etc. – on/off, PID, open loop, etc.

Annex:

a)

```
#include <Arduino.h>
#include "L298N.h"
#include "LAMP.h"
#include "LiquidCrystal.h"
#include "mystdio.h"
#include "DHT.h"
#include "PID_v1.h"

#define DHTPIN 2
#define DHTTYPE    DHT22

DHT dht(DHTPIN, DHT22);
L298N motor(A1, 6, 5);
LAMP lamp(7);
LiquidCrystal lcd(10);
Mystdio mystdio;

int humiditySetpoint;

//Define Variables we'll be connecting to
double Input, Output, motorSetpoint;
int msp;
//Specify the links and initial tuning parameters
double Kp=1, Ki=0, Kd=0;
PID myPID(&Input, &Output, &motorSetpoint, Kp, Ki, Kd, DIRECT);

void setup() {
  mystdio.open(StreamIO::SERIALIO);

  motor.setSpeed(200);
  motor.forward();

  lcd.begin(16, 2);
  printf("Started \r");

  lcd.clear();
  lcd.print("Reading setpoints");

  printf("Humidity setpoint: ");
  scanf("%d", &humiditySetpoint);
  // printf("Motor speed setpoint: ");
  // scanf("%d", &msp);
  // motorSetpoint = (double)msp;

  lcd.print("Started");

  //turn the PID on
```

```
  myPID.SetMode(AUTOMATIC);

  dht.begin();
}


void loop() {
  delay(1000);
  int humidity = dht.readHumidity();
  lcd.clear();
  lcd.print("hum:");
  lcd.print(humidity);
  lcd.print("sp:");
  lcd.print(humiditySetpoint);


  if(humidity > humiditySetpoint) {
    lamp.setOn();
  } else {
    lamp.setOff();
  }
}
```