

GREENHOUSE

Team members

- Téo Dos Santos
- Gabriel Taffin
- Alexandre Truong
- Renaud Visioli

Link to this file in your GitHub repository:

<https://github.com/teods95/DigitalElectronics-2/tree/main/PROJECT>

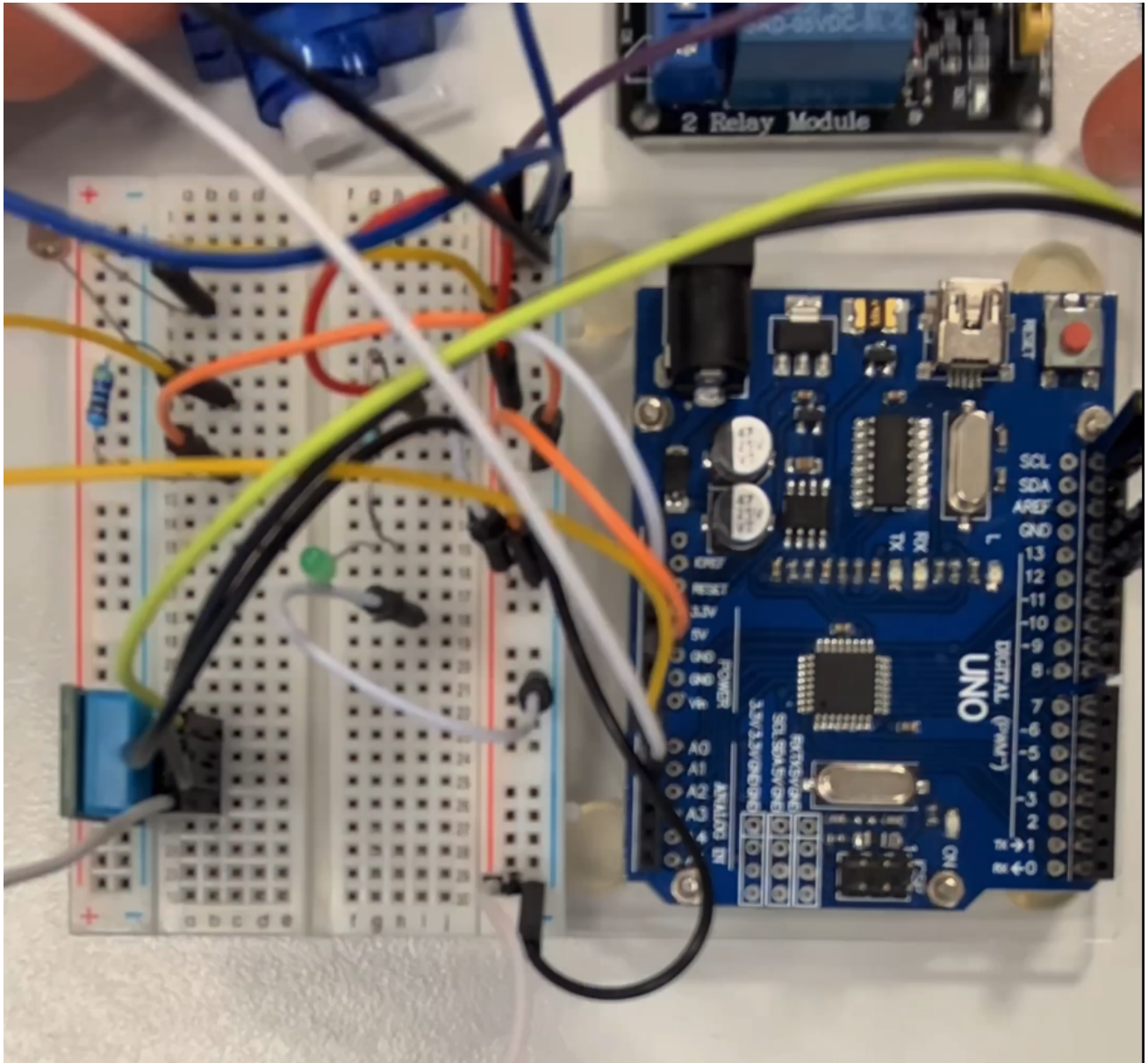
Table of contents

- [GREENHOUSE](#)
 - [Team members](#)
 - [Table of contents](#)
 - [Project objectives](#)
 - [Hardware description](#)
 - [Libraries description](#)
 - [Main application](#)
 - [Video](#)
 - [References](#)

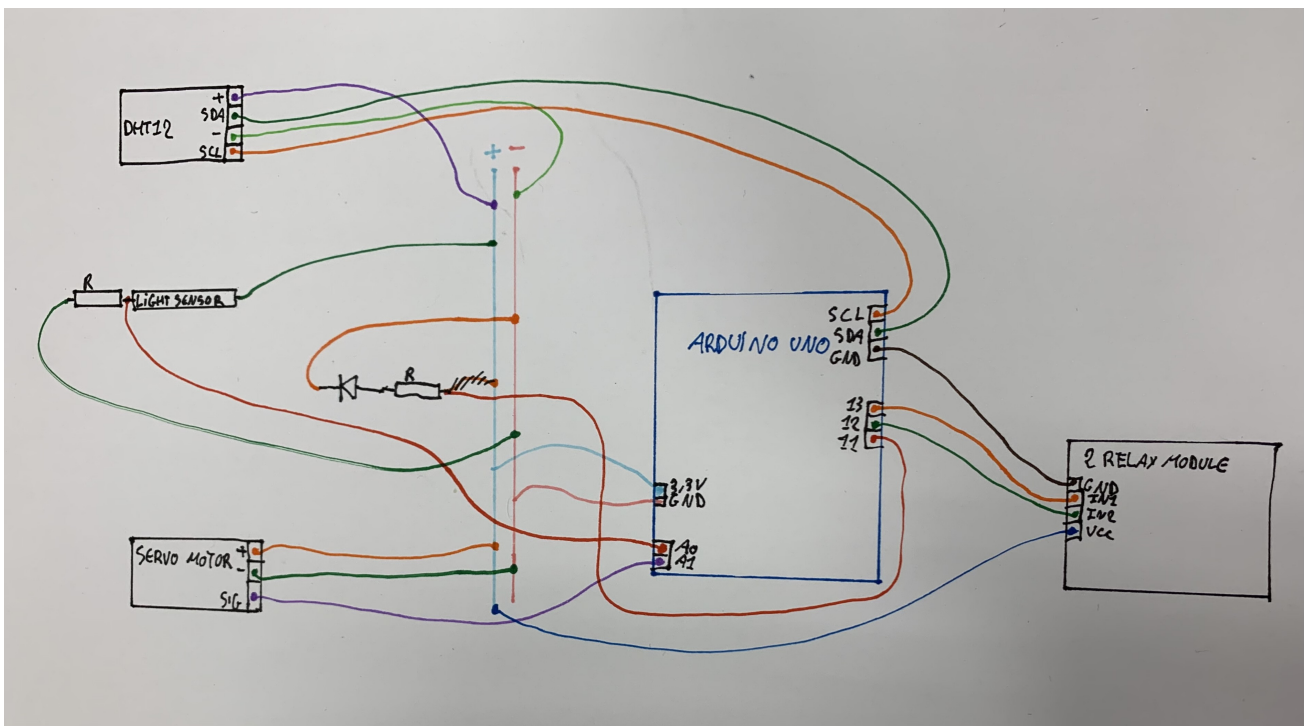
Project objectives

Make a monitoring and control system of a GreenHouse, including : light, moisture and temperature sensor and also : irrigation pump control, DC fans and windows opening or closing.

Hardware description



- Arduino Uno board : ATmega328P
- DHT12 Digital temperature and humidity sensor
- Photoresistor : Light sensor
- Resistor 330 Ohm x 2
- LED
- Relay Modules
- SG90 Micro-servo motor

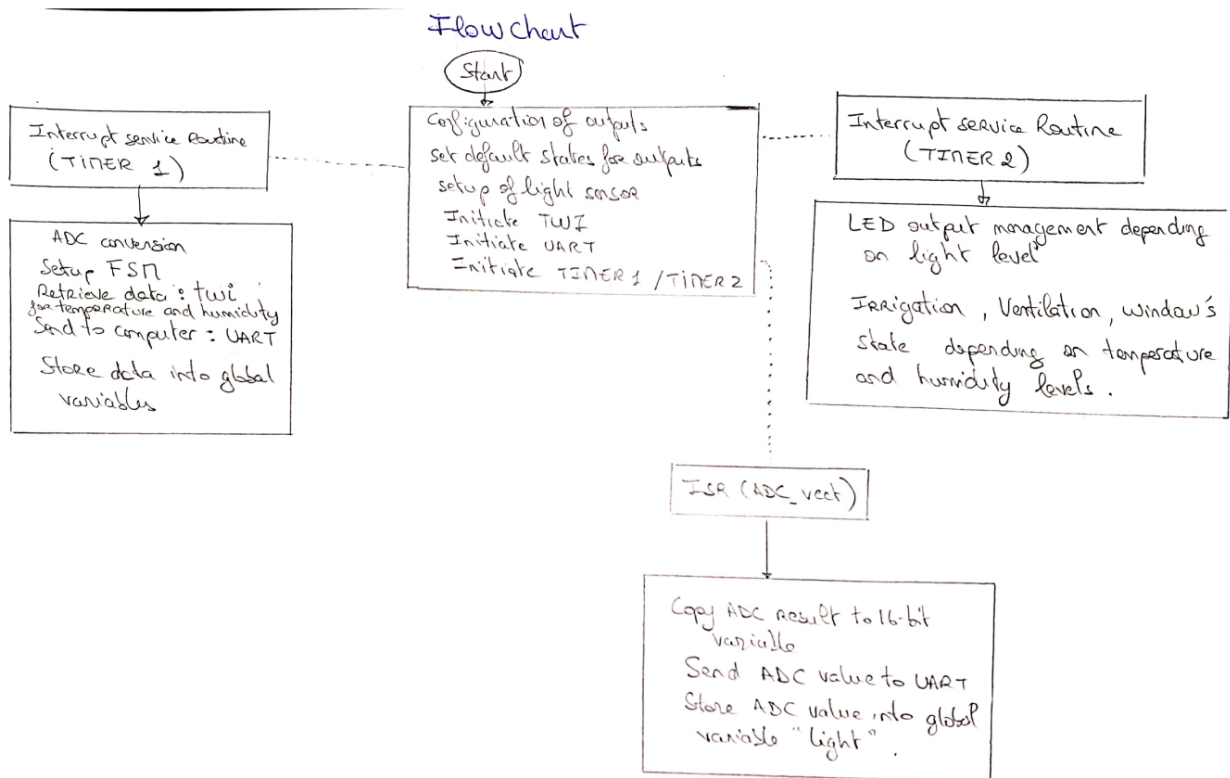


Libraries description

The libraries used, are libraries used in previous labs :

- `avr/io.h` : AVR device-specific IO definitions
- `avr/interrupt.h` : Interrupts standard C library for AVR-GCC
- `stdbool.h` : Boolean type and values
- `timer.h` : Timer library for AVR-GCC
- `tdlib.h` : C library. Needed for conversion function
- `uart.h` : Peter Fleury's UART library
- `twi.h` : TWI library for AVR-GCC
- `gpio.h` : Contains functions for controlling AVR's gpio pin(s).
- `util/delay.h` : Contains delay functions

Main application



Scanned avec CamScanner

Our project consists of the operation of an autonomous greenhouse. Our greenhouse has sensors: a light sensor, a temperature sensor and a humidity sensor. The objective of our project is to be able to control these variables in real time so that they remain in optimal intervals for the culture of plants. Therefore, depending on the values of the variables, actions can be taken in order to try to keep the variables in the optimal ranges. Irrigation, opening the window or turning on the fans can be activated.

For this we have considered the optimal value intervals for humidity, temperature and light, and we have defined the set of actions to be performed for each combination of divergence from these intervals.

We realized a dynamic system thanks to the use of two separate interrupts: one realizing the recovery of the sensors' data, while the other one realizes the management of the outputs according to the retrieved data. Both running in parallel, this allows us to act on the outputs of the system without ceasing to update the sensor data.

PS : We did not have enough time to complete the project during the lab session. Therefore, our complete system could only be partially tested: we were able to test the operation of each output separately as well as the recovery of sensor data. Nevertheless, we are not sure that the outputs of our system behave according to our expectations in relation to the variations of the data recovered by the sensors.

Video

<https://drive.google.com/file/d/1YLNULn7bUI6V9HJiC57NfYxINMQx4-yd/view?usp=sharing>

References

<https://moodle.vut.cz/course/view.php?id=241310>

<https://github.com/tomas-fryza/Digital-electronics-2>

http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf

<https://www.handsontec.com/dataspecs/module/2Ch-relay.pdf>