Arduino: Digital clock with temperature measurement

Device description

The project uses an Arduino and the following components:

- **RTC_DS1307** - Real-time clock module (RTClib library),
- **TM1637Display** - 4-digit display with backlight - controlled by CLK and DIO,
- **DHT11** - Temperature and humidity sensor (connected to pin 9),
- **DS18B20** - digital thermometer (sensors connected to OneWire bus on pin 3).

Library import

- `RTClib.h` - RTC clock support,
- `Arduino.h` - Arduino platform base functions,
- `TM1637Display.h` - 4-digit display control,
- `DHT.h` - DHT11 sensor control,
- `OneWire.h` i `DallasTemperature.h` - DS18B20 support,
- `SPI.h`, `Wire.h` - communication libraries (Wire™ used by RTC).

Pin definitions and objects

- `CLK 6`, `DIO 4` - TM1637 display pins,
- `DHTPIN 9`, `DHTTYPE DHT11` - DHT11 sensor configuration,
- `ONE_WIRE_BUS 3` - bus for DS18B20.
- Objects:
  - `RTC_DS1307 rtc;`
  - `TM1637Display display(CLK, DIO);`
  - `DHT dht(DHTPIN, DHTTYPE);`
  - `OneWire oneWire(ONE_WIRE_BUS);`
  - `DallasTemperature sensors(&oneWire);`

setup()

1. `rtc.begin()` - starts the RTC clock,
2. `sensors.begin()` - initialises the DS18B20,
3. sets the brightness of the display (`setBrightness`) and shows „8888” as a test,
4. Initialises the serial port (`Serial.begin`) - to debug measurements,
5. `dht.begin()` - initialises the DHT11 sensor.

loop() - main cycle

1. `sensors.requestTemperatures()` i `getTempCByIndex(0)` - DS18B20 temperature

reading,
2. conversion to `int` - rounding (`round`),
3. `rtc.now()` - gets the current time; extracts the hour and minute,
4. read from DHT11: `readHumidity()` i `readTemperature()`,
5. Formatting the time as a 4-digit string (e.g. „0835”),
6. Converting to a number (`toInt()`) and displaying sequentially:
    - hour and minute,
    - temperature from DHT11 (°C),
    - humidity from DHT11 (%),
    - temperature from DS18B20 (total value),

Each reading - 2 seconds.

Applied modules/libraries

- **RTClib** - Support for DS1307/DS3231 RTC modules,
- **TM1637Display** - showing digits on 4-segment display with decimal point,
- **DHT** - humidity and temperature reading from the DHT11,
- **OneWire + DallasTemperature** - support of even multiple 1-wire thermometers,
- **SPI** i **Wire** - required by some of the above (e.g. RTC).

Example of operation

1. At start-up, the display shows „8888” - operation test of all segments,
2. Then it displays the time (hhmm),
3. Then sequentially every 2 seconds:
    - temperature DHT11 (e.g. „23.5” → „23.5°C”),
    - DHT11 humidity (e.g. „45.2” → „45.2%”),
    - DS18B20 temperature (e.g. „24” → 24°C).

Possible modifications

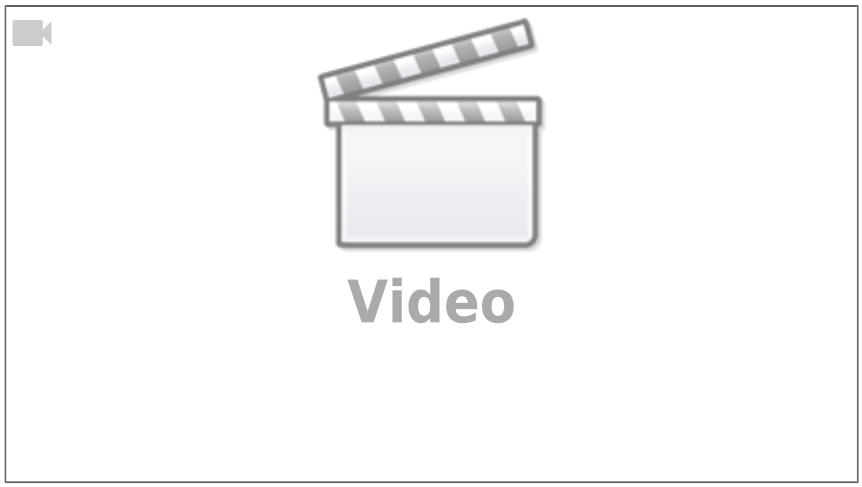
- Addition of buttons for time setting,
- Data registration to EEPROM or SD card,
- Extension of the interface - e.g. menu selection of displayed values,
- Use of DS3231 for greater RTC accuracy.

Summary

The device is a 'digital' type clock with temperature and humidity measurement, visualising the values on a TM1637 display. It uses three different sensors and simple Arduino libraries, making the project a great example of hardware and software integration in a single system.

Presentation video

Video recorded a long time ago and in English, so please be understanding ;)



Code

```
#include ''RTClib.h''
#include <Arduino.h>
#include <TM1637Display.h>
#include ''DHT.h''
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SPI.h>
#include <Wire.h>

#define CLK 6
#define DIO 4
#define DHTPIN 9
#define DHTTYPE DHT11

#define ONE_WIRE_BUS 3

RTC_DS1307 rtc;

TM1637Display display(CLK, DIO);

DHT dht(DHTPIN, DHTTYPE);

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

void setup() {

  rtc.begin();
  sensors.begin();

  display.setBrightness(0x0f);
  display.showNumberDecEx(8888,0b01000000, false);
  delay(3000);
  display.clear();
  display.setBrightness(3);
```

```
  Serial.begin(57600);
  dht.begin();
}

void loop() {
  sensors.requestTemperatures();
  float tempC = sensors.getTempCByIndex(0);
  int tempINT = round(tempC);
  DateTime now = rtc.now();
  int Minute = now.minute();
  int Hour = now.hour();
  String strMinute;
  String strHour;
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  if(Hour < 10)
    {strHour = ''0'' + String(Hour);}
  else if(Hour >= 10)
    {strHour = String(Hour);}
  else if(Hour == 0)
    {strHour = ''00'';}

  if(Minute < 10)
    {strMinute = ''0'' + String(Minute);}
  else if(Minute >= 10)
    {strMinute = String(Minute);}
  else if(Minute == 0)
    {strMinute = ''00'';}



  String Time;
  Time += strHour;
  Time += strMinute;

  if(Time == ''0''){
    Time = ''0000'';
  }

  Serial.println(tempC);
  Serial.println(tempINT);
  int numTime = Time.toInt();
  display.clear();
  display.showNumberDecEx(numTime,0b01000000, true);
  delay(2000);
  display.clear();
  display.showNumberDec(t,false,2,1);
  delay(2000);
  display.clear();
  display.showNumberDec(h,false,2,1);
```

```
  delay(2000);
  display.clear();
  display.showNumberDec(tempINT,false,2,1);
  delay(2000);


}
```