



Lab 05, Lab 08 & Lab 09 : Tutorial

Design and implement our embedded system

GROUP NE 2

STROMITZKY Pierre
SIMONIN Corentin
SEZER Ayşe-Nur
MONGIN Lucas

Embedded systems : Architecture & Programming

27/03/2018

Table of contents

Introduction	2
Our Embedded System	3
Global description	3
What do you need for this project ?	4
The electronics components :	4
What are these components and how their are running?	4
Why do we use these components ?	5
Diagram	6
A more understable sketch with Fritzing:	8
The Tutorial !	9
Let's do our arduino !	9
The TEMBOO thing !	12
Web Programmation part	14
Introduction	14
Code	14
Sources :	16

Introduction

Nowadays, our environment changes quickly: light, temperature, wind, time... all those things pass by in no time. Thus we have never been more connected than today, embedded systems are everywhere and changes lives. That is why, for learning purpose, we are creating an embedded system; not really useful by itself but which can represent really well what an embedded system is, so it's a good study case, and an opportunity for us.

Our Embedded System

Global description

During this tutorial, we'll explain :

- how to vary the brightness of a first led according to temperature variations
- how to vary the brightness of a second led inversely to brightness variations
- how to send emails with current temperature informations thanks to Arduino
- how to display data from a database on a webpage

Furthermore, we wanted to show you how to display current temperature informations in a web page using Arduino.

Unfortunately, we didn't manage to meet the deadlines for this task. Therefore, we will carry out another tutorial as soon as possible. Nevertheless, if you don't have the time to wait, we suggest you the following youtube tutorial: <https://www.youtube.com/watch?v=o05ltCWC2kk&t=2s>

What do you need for this project ?

The electronics components :

- Arduino Yùn
- 1 White LED (for brightness)
- 1 red LED (for temperature)
- 1 TMP36 (for temperature)
- LDR Photoresistor 1Mohm (for the brightness)
- 3 Resistors:

- 2 for the LED: 100 ohm
- One for the Photoresistor : 10Kohms
- Your computer
- Some wires
- A cable for connection
- And most important, your good mood ;)

What are these components and how do they run ?

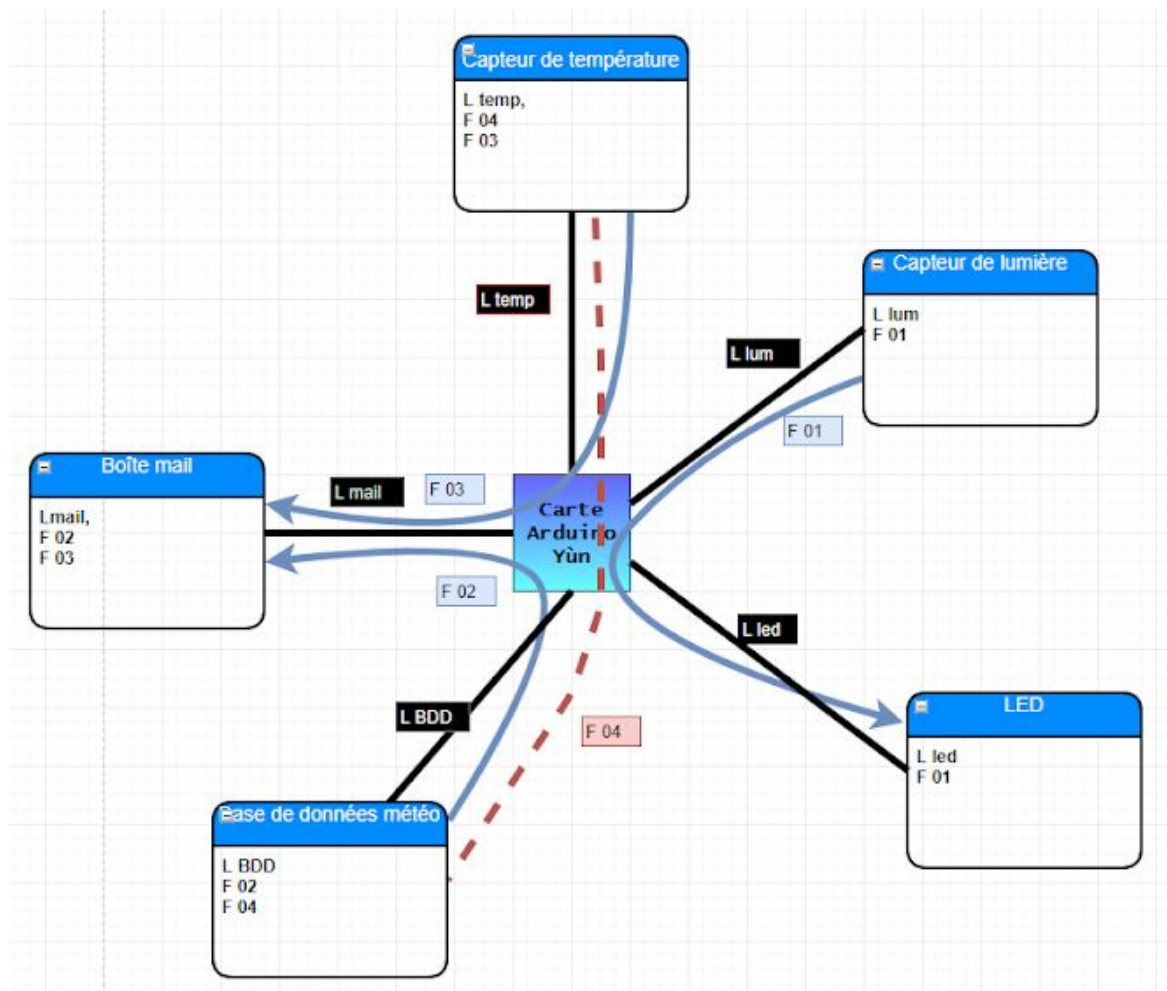
- Photoresistor:
 - It is like a resistor which his value change according to the light it receives
 - It exists different type of photo-resistor, having different resistor values depending on the ambient brightness
 - its resistance decrease when the photoresistor is illuminated so we can use it in order to measure the ambient brightness.
- TMP36:
 - it is a sensor that comes in the form of a small transistor
 - it can measure temperature from -50 to 150°C with a precision of $\pm 0.5^{\circ}\text{C}$
 - it exists equivalent type of sensor in order to measure the ambient temperature, having different range for the measure of temperature and their precision are different too, in addition, the code could be different if we use another type.
- We find that we needed to use there resistor thanks to the famous "U=RI" Ohm's law equation, we cannot use to low resistances because we don't want to burn our main board.

Why do we use these components ?

- Photoresistor:
 - Ambient brightness measurement in order to adjust lightness
- TMP36:
 - A lot of advantage: the coverage of a wide temperature range (from -50° to 150°C) and the output voltage is totally independent of the power supply of the sensor.

Diagram

We can figure it out better with a diagram like that:



This diagram represents the interactions inside of the system, between the sensors and the functionalities.

I admit that this diagram can be quite hard to understand like that so we can summarise the thinks that our system can do with few lines :

It will bright inversely proportional to the brightness of the surroundings , like an adaptive light bulb with a white LED. And It will bright inversely proportional to the temperature of the surroundings with a red LED

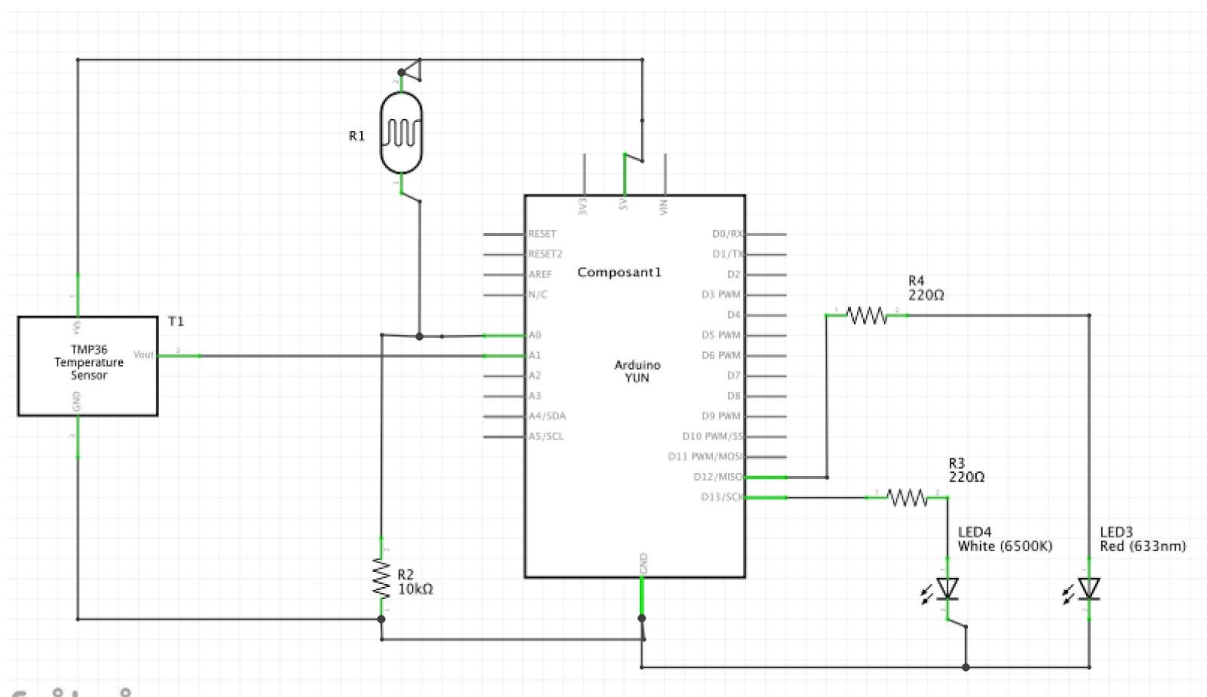
It will tell to the user the temperature of the place, but exactly one year before the time of the measure, for that the system will be connected to internet and a database

Then it will also compare the difference between the actual temperature and the one of the precedent year

All the data that the user will need will be send by mail by the system.

A more understable sketch with Fritzing:

- Schematic view of the assembly:



The Tutorial !

Okey, now you should have understood the concept, we should go practice ! and with our help (with love) do it yourself !

“yey now i can do some code and stuff”

NO ! i'm sorry, but we are talking about an embedded system, so we have to get rid of this wire !

So we will connect ourself with internet with the arduino yun board !

let's do it, but arduino explains better than everybody how to do it, so this is their tuto : <https://www.arduino.cc/en/Guide/ArduinoYun>

go to :

Configuring the onboard WiFi

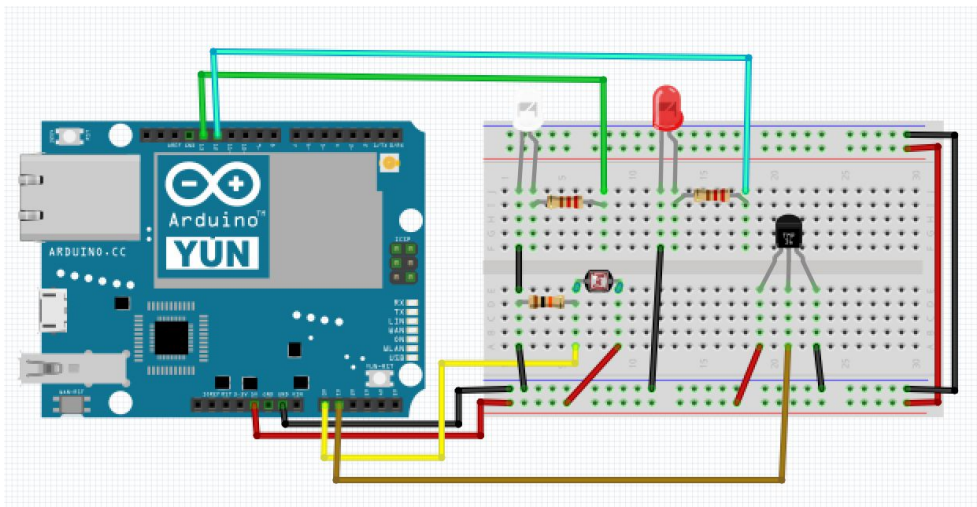
and connect the yun !

It's done ? fine let's go !

Let's do our arduino !

First of all, let's do connections with the arduino board, so we can have all the functionalities of the system.

Just follow this assembly :



Simple for the moment isn't it ?

Be advised, it will not be that simple ...

So, we should open the code now !

The code is in the document : [CODE.ino](#)

and the [TembooAccount.h](#) one, we will talk about it later

```
// Invoke the Temboo client
SendEmailChoreo.begin();

// Set Temboo account credentials
SendEmailChoreo.setAccountName(TEMBOO_ACCOUNT);
SendEmailChoreo.setAppName(TEMBOO_APP_KEY_NAME);
SendEmailChoreo.setAppKey(TEMBOO_APP_KEY);

// Set Choreo inputs
SendEmailChoreo.addInput("FromAddress", "XXXXXXXXXX");
SendEmailChoreo.addInput("Username", "XXXXXXXXXX");
SendEmailChoreo.addInput("Subject", "I HAVE A TEMPERATURE TO TELL YOU");
SendEmailChoreo.addInput("ToAddress", "XXXXXXXXXX");
SendEmailChoreo.addInput("MessageBody", message );
SendEmailChoreo.addInput("Password", "XXXXXXXXXX");
```

"WAIT WAIT WAIT, i know arduino but what the hell is "SendEmailChoreo" or Temboo ?"

Yeah, i understand, but don't be too speedy my gonzales. It's for later. First we should just do the LED and captors things, OK ?

```
void loop() {
  photoValue = analogRead(photoPin);
  photoValue = map (photoValue, photoMin, photoMax, 0, 255);
  photoValue = constrain(photoValue, 0, 255);

  tmpValue = analogRead(tmpPin);
  float volt = tmpValue * 5000.0 / 1024.0;
  float tmp = (volt - 500) / 10;

  String tmpSt=String(tmp);
  String message = String("I have something to tell you: " + tmpSt + " °C");

  ledvertBrightness = photoValue;
  ledrougeBrightness = map(tmpValue, -50, 150, 0, 255); //tmp: couverture de -50 à 150°C)

  digitalWrite(ledvertPin, HIGH);
  //digitalWrite(ledrougePin, ledrougeBrightness);
  //analogWrite(ledvertPin, ledvertBrightness);
  //analogWrite(ledrougePin, ledrougeBrightness);

  Serial.print("Valeur de la photorésistance: ");
  Serial.println(photoValue);
  Serial.print("Valeur de la température: ");
  Serial.print(tmp);
  Serial.println("°C");
  delay(1000);
}
```

So, look at this loop, we take the data from the sensors (be sure to take the goods pins like in this code little guy)

Then the code use the mysterious TEMBOO thing (we will see that later)

And we adjust the led in function of the sensors !

The TEMBOO thing !

So now we will talk about temboo !

“yea, yea, so what’s that ?”

Temboo is a wonderful tool for doing specials stuff when we are connected with internet, in our case, send a mail ! WOW !

we just need to send the data to temboo following a library and temboo will use our gmail account and send the data for us !

But before that, You need to set temboo to use it, young padawan !

And, i’m sorry but for this part i will not be your jedi master :/

Temboo itself have really great tutorials for that ! so we will use it (and it’s also because i’m a lazy bastard) Here, take the URL of the tuto :

<https://temboo.com/arduino/yun/send-an-email>

Follow it well, step by step... i will wait until you finished !

...

so

...

it’s quite long

...

Have you finished ?

Yes ?

Finally !

Okey, so together we will be sure that all is green,

You should have 2 document in the folder of the project !

The one with all the normal code

and one strange one named “**TembooAccount.h**”

so the second one will have all the data of your temboo account, the way it should be filled is presented in the tuto of temboo, if it’s not filled, that’s

because you are playing with me ! you didn't follow the tuto ! If you followed it, we have a major issue, so I have the pleasure to announce you that in this case you will have the pleasure to start it again !

The most important part of the code that you will have to adapt is this one

```
// Invoke the Temboo client
SendEmailChoreo.begin();

// Set Temboo account credentials
SendEmailChoreo.setAccountName(TEMBOO_ACCOUNT);
SendEmailChoreo.setAppKeyName(TEMBOO_APP_KEY_NAME);
SendEmailChoreo.setAppKey(TEMBOO_APP_KEY);

// Set Choreo inputs
SendEmailChoreo.addInput("FromAddress", "XXXXXXXXXXXXXXXXXXXX");
SendEmailChoreo.addInput("Username", "XXXXXXXXXXXXXXXXXXXX");
SendEmailChoreo.addInput("Subject", "I HAVE A TEMPERATURE TO TELL YOU");
SendEmailChoreo.addInput("ToAddress", "XXXXXXXXXXXXXXXXXXXX");
SendEmailChoreo.addInput("MessageBody", message );
SendEmailChoreo.addInput("Password", "XXXXXXXXXXXXXXXXXXXX");
```

So on the image at the bottom :

In the part "FromAdress" you will put the sender of the mail like "arduinoyun@gmail.com" (don't forget that you need a gmail address)

in the part "Username" you will put the name that you have put in temboo

In the part "Subject" you will put the title of the mail

In the part "ToAddress" you will put the recipient of the mail like "lovingmommy@hotmail.com"

In the part "MessageBody", we will put the content of the message, we had put it in a variable before, for simplicity

Finally in the part "Password" You have to put your passport !!

"Yea... but witch one ?"

Your gmail password of your beautiful "arduinoyun@gmail.com" adress

Web Programmation part

Introduction

Why not using the web to see the measured temperature with the Arduino card and to compare it with real data ? In this part, you will learn to do so very easily.

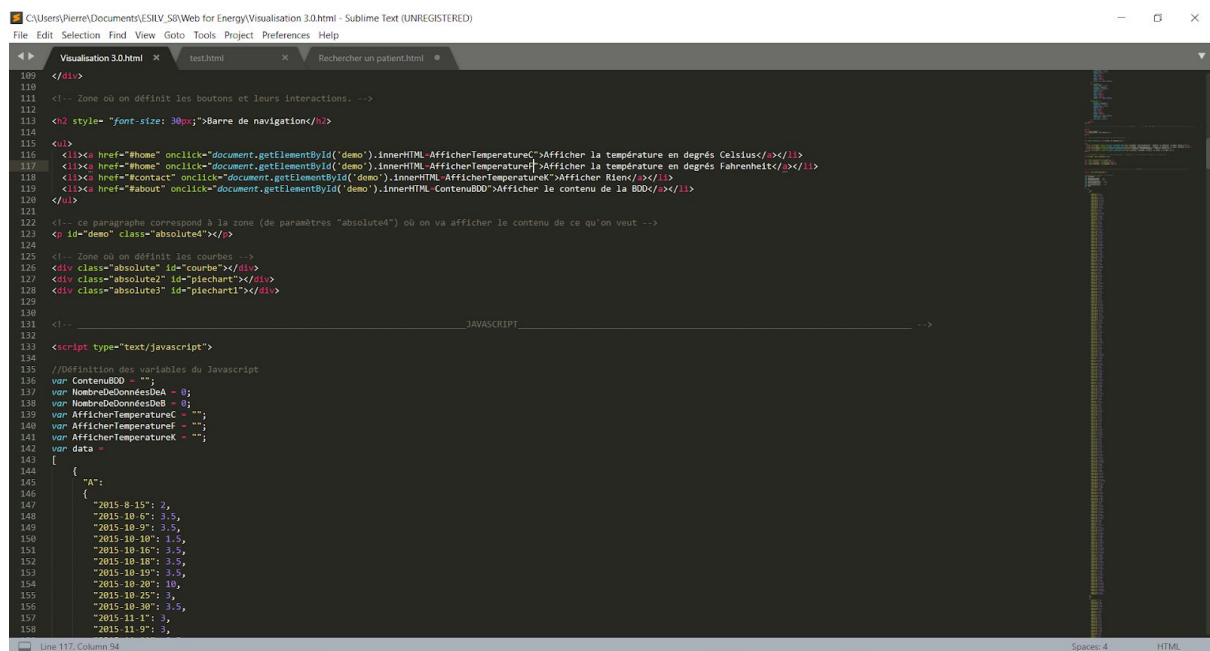
Code

First, you have to install a text editor if you don't have one install.

You may use Sublime Text (URL :<https://www.sublimetext.com/>), that provides clear visibility for your code. You will use HTML and Javascript encoding.



This is an idea of what it looks like in real :



```

109 </div>
110
111 <!-- Zone où on définit les boutons et leurs interactions. -->
112
113 <div style="font-size: 30px;">Barre de navigation</div>
114
115 <div>
116 <div href="#" onclick="document.getElementById('demo').innerHTML=AfficherTemperature">Afficher la température en degrés Celsius</div>
117 <div href="#" onclick="document.getElementById('demo').innerHTML=AfficherTemperature">Afficher la température en degrés Fahrenheit</div>
118 <div href="#" onclick="document.getElementById('demo').innerHTML=AfficherTemperature">Afficher Rien</div>
119 <div href="#" onclick="document.getElementById('demo').innerHTML=ContenuBDD">Afficher le contenu de la BDD</div>
120 </div>
121
122 <!-- ce paragraphe correspond à la zone (de paramètres "absolute4") où on va afficher le contenu de ce qu'on veut -->
123 <div id="demo" class="absolute4"></div>
124
125 <!-- Zone où on définit les courbes -->
126 <div class="absolute" id="courbe"></div>
127 <div class="absolute2" id="piechart"></div>
128 <div class="absolute3" id="piechart1"></div>
129
130
131 <!-- JAVASCRIPT -->
132
133 <script type="text/javascript">
134
135 //Définition des variables du Javascript
136 var ContenuBDD = "";
137 var NombreDonnéesDeA = 0;
138 var NombreDonnéesDeB = 0;
139 var AfficherTemperatureC = "";
140 var AfficherTemperatureF = "";
141 var AfficherTemperatureK = "";
142 var data =
143 [
144 {
145 "A":
146 {
147 "2015-8-15": 3,
148 "2015-10-6": 3.5,
149 "2015-10-9": 3.5,
150 "2015-10-10": 1.5,
151 "2015-10-16": 3.5,
152 "2015-10-18": 3.5,
153 "2015-10-19": 3.5,
154 "2015-10-20": 10,
155 "2015-10-25": 3,
156 "2015-10-30": 3.5,
157 "2015-11-1": 3,
158 "2015-11-9": 3,

```


Beautiful ? Sure ! Now, let's code some lines :

At first, let's define the head and the title of the page :

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="utf-8">
5      <title>What is the temperature ?</title>
6    </head>
```

Just add the meta and a title of your choice for the solution.

Now, let's define the zones in which you would like to show your stuff : define a text zone you name "temperature" and another one in which you will add an ID, because we want to access it, in order to show the temperature :

```
8    <body>
9      <p>Temperature :</p>
10     <p id="demo"></p>
```

Now, take a look to this site : <https://www.apixu.com/>

It can provide data about the meteo of a specific area. Just select the area you want and the period, and the site will provide you data.

You can choose the extension of the file. We guest you to choose the .json one.

Now, you will get access to a very messy web page. Copy its link and paste it on a "requestURL" variable. Line 14 is useful to make an HttpRequest, needed for what is following. This request will open (line 15), read the file as a text (line 16), and attract the data (line 17).

```
12    <script>
13      var requestURL = 'http://api.apixu.com
14      var request = new XMLHttpRequest();
15      request.open('GET', requestURL);
16      request.responseType = 'text';
17      request.send();
```

And then, the function is here to say "hey, my variable will be the response of the site (that we read as a text, remember) and I will store it as a JSON (line 21, JSON.parse). And I want to get, in that JSON, the temperature, so i get the "demo zone" that we defined before, and i set

its innerHTML (his identity, if you mind) to the address of the temperature. And ... That's it

```
18     request.onload = function()
19     {
20         var variable = request.response;
21         var JsonObjMonContenu = JSON.parse(variable);
22         document.getElementById("demo").innerHTML = JsonObjMonContenu.forecast.forecastday[0].day.avgtemp_f;
23     }
24     </script>
25 </body>
26 </html>
```

Of course, you can add some design to this very simple code, to make it really cool to look at.

Just because I am nice, I give you a beautiful designed code ! =) It will be in the attached file.

Sources :

- <https://openclassrooms.com/courses/programmez-vos-premiers-montages-avec-arduino/les-capteurs-electroniques>
- <http://www.manuel-esteban.com/arduino-capteur-de-luminosite/>
- https://fr.wikipedia.org/wiki/Lux_%28unit%C3%A9%29
- <http://forum.arduino.cc/index.php?topic=272492.0>
- <https://temboo.com/arduino/yun/send-an-email>
- <https://github.com/Polyfe/PoC-Take-And-Save-Data-Project>