A patient's temperature remote control system based on NODEMCU ESP8266

Anas El Kasmi[1]**,** *Mostafa* Abouricha **[2], ** and *Abdelkader* Boulezhar [3]
[1] LRESD, Department of Physics, Faculty of Sciences-AÏn Chock, Hassan II University of Casablanca, BP 20100, Morocco
[2] LMTI, Department of Physics, Faculty of Sciences Ibn Zohr University of Agadir BP 8106, Morocco.
[3] LRESD, Department of Physics, Faculty of Sciences-AÏn Chock, Hassan II University of Casablanca, BP 20100, Morocco

Abstract. With the prompt development in wireless communication, in addition to the evolution of wireless patient monitoring system, and to reduce the death of a big number of patients who endear as a result of heart failure or sudden high temperature in children. This paper proposed a sophisticated method of observation human body parameters like temperature measurements with employ at NodeMcu ESP8266 unit in the Internet of Things (IoT). This prototype is based on NodeMcu module (a static access point that provides the WIFI network, a server, a client and a mobile access point attached to the remote surveillance object) programmed under Arduino IDE and communicating between them via the HTTP protocol.

1 INTRODUCTION

The development of the medical field largely resulted from the rapid development of technology. This has led to an increase in the average age of human beings and low mortality rates, especially in industrialized and developed countries [1]. The lack of modern medical equipment that helps diagnose and treat patients has led to poor medical services in developing countries and weak technology such as Third World countries .

Today, due to advances in technology and the Internet of Things, step-by-step tasks such as mailing, accessing bank records, monitoring the degree of other tasks [1], eHealth [2], smart home [3-4], smart city [5], intelligent transport [6], wireless sensor networks [7], smart grid [8], objects of remote monitoring systems [9]. However for any IoT application[10].

Here in this project, we will create an IoT based health monitoring system that records body temperature and also send an email alert. Readings are recorded in the ThingSpeak and Google Sheets so that the patient's health can be monitored from anywhere in the world via the Internet.

Note that the communication between the different nodes is done by the HTTP protocol of the library ESP8266.h and the integrated development environment Arduino_IDE.

The rest of the paper is organized as follows. In section 2 related works were defined. In section 3 proposed system were explained.. In section 4 Hardware realizations were shown. In section 5 System implementation and the conclusion and future work was discussed in section 6.

2 NODEMCU (ESP8266)

The NodeMcu ESP8266 includes a WiFi module, memory, a serial link and manages GPIO ports. This microcontroller has the ability to perform activities related to WIFI.

In this article, we will discuss the use of the NodeMcu ESP8266 unit in the Internet of Things, (IoT) using AT or Arduino IDE commands for data exchange.

-802.11 b / g / n wireless standard

-Support three STA / AP / STA + AP working modes

-Built-in TCP / IP protocol stack to support multiple TCP client connections (5 MAX)

-AD0: 1 ADC channel

-Power input: 4.5V - 9V (10VMAX), powered by USB \

-Transfer rate: 110-460800bps

-Support UART / GPIO data communication interface

-Remote Firmware Upgrade (OTA:Over-the-air programming)

-Smart Link Smart Grid Support \

Fig. 1. The image and main characteristics of the NodeMcu used

Pinout and connectivity :

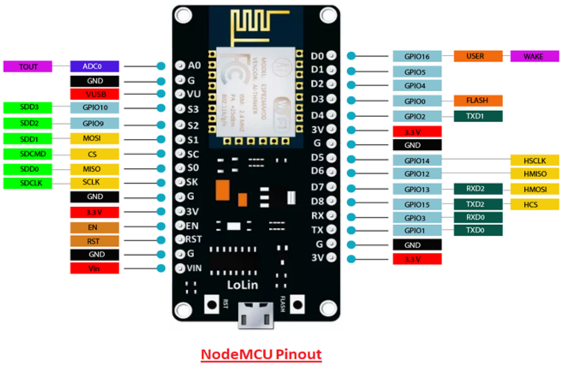


Fig. 2. Pinout and connections of the NodeMcu card

2.1 LM35 temperature sensor

This component makes it possible to measure the temperature around the sensor. It is precise, inexpensive, very easy to use and extremely reliable. The temperature measured by the sensor is proportional to the voltage on the analog output of the sensor. To reduce the difference between the actual temperature and the measured temperature, especially at extremes (around -55 ° and around 150 °).

2.1.1 Features

- Calibrated Directly in Celsius (Centigrade)
- Linear + 10-mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at 25°C)
- Rated for Full −55°C to 150°C Range
- Suitable for Remote Applications
- Low-Cost Due to Wafer-Level Trimming
- Operates From 4 V to 30 V
- Less Than 60-µA Current Drain
- Low Self-Heating, 0.08°C in Still Air
- Non-Linearity Only ±¼°C Typical
- Low-Impedance Output, 0.1 Ω for 1-mA Load

2.1.2 Applicaions

Power Supplies
- Battery Management
- HVAC
- Appliances

2.1.3 Description

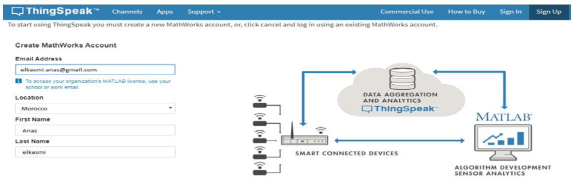
The LM35 series are precision integrated-circuit temperature devices with an output voltage linearly proportional to the Centigrade temperature. The LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, linear output, and precise inherent calibration of the LM35 device makes interfacing to readout or control circuitry especially easy. The device is used with single power supplies, or with plus and minus supplies. As the LM35 device draws only 60 µA from the supply, it has very low self-heating of less than 0.1°C in still air. The LM35 device is rated to operate over a −55°C to 150°C temperature range, while the LM35C device is rated for a −40°C to 110°C range (−10° with improved accuracy). The LM35-series devices are available packaged in hermetic TO transistor packages, while the LM35C, LM35CA, and LM35D devices are available in the plastic TO-92 transistor package. The LM35D device is available in an 8-lead surface-mount small-outline package and a plastic TO-220 package.

2.2. ThingSpeak

ThingSpeak provides a great tool for IoT based projects. With ThingSpeak, we can monitor our data and control an online system, using the web pages provided by ThingSpeak, we can collect sensor data, analyze it, visualize it and finally trigger actions.

ThingSpeak will be used to monitor temperature online over the Internet. For this, it is necessary to set up the IFTTT platform to connect ThingSpeak to the messaging service so that the alert message can be sent whenever the patient is in a critical state.

<u>1st step</u>: Creation of a user account on ThingSpeak.com.



To start using ThingSpeak you must create a new MathWorks account, or, click cancel and log in using an existing MathWorks account.

Create MathWorks Account

Email Address: elkasmi.anas@gmail.com

To access your organization's MATLAB license, use your school or work email.

Location: Morocco

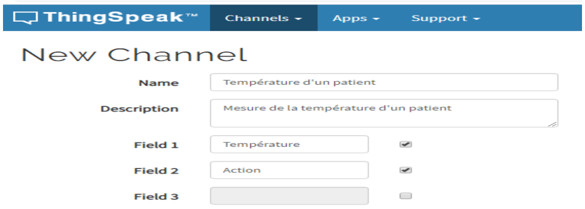
First Name: Anas

Last Name: elkasmi

Fig. 3. Window to create a count on Thingspeak.com site

https://doi.org/10.1051/e3sconf/202129701053

2nd step: In the "Channels" menu and click on the New channel option on the same page to create a new channel.

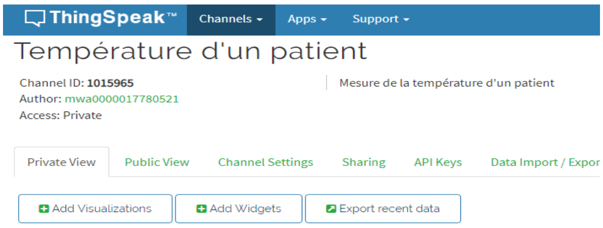


New Channel

Name: Température d'un patient
Description: Mesure de la température d'un patient
Field 1: Température ☑
Field 2: Action ☑
Field 3: □

Fig. 4. Window to create channel and fields on Thingspeak.com site

3rd Step: Now you will see a template for channel creation, fill in the name and description as you want. Fill in "Temperature" and "Action" in the fields Field 1, Field 2, check the boxes of the fields. Finally, check Save channel. Now your new channel has been created



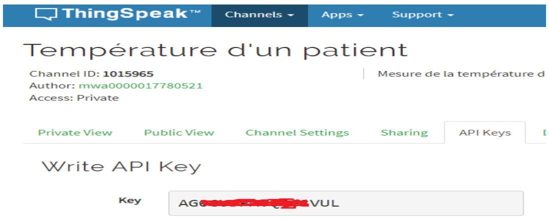
Température d'un patient

Channel ID: 1015965
Author: mwa0000017780521
Access: Private

Mesure de la température d'un patient

Fig. 5. Window's name of channel and action to generate

4th Step: You will see three graphics as shown below. Note the write API key and we'll use that key in our code.



Température d'un patient

Channel ID: 1015965
Author: mwa0000017780521
Access: Private

Mesure de la température d

Write API Key

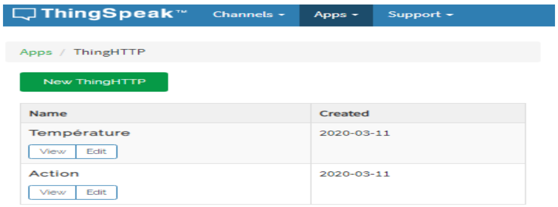
Key: AG[redacted]VUL

Fig. 6. Window to show write API key on the channel

5th Step: Now we will use the ThingHTTP app on the server to launch the IFTTT app to enter data into Google Sheets and send emails / SMS

2.3. ThingHTTP and IFTTT

ThingHTTP enables communication between devices, websites, and web services without having to implement the protocol at the device level. You can specify actions in ThingHTTP, which you want to perform with other ThingSpeak apps such as React. To make ThingHTTP new, we'll need a URL to run which we'll get from IFTTT

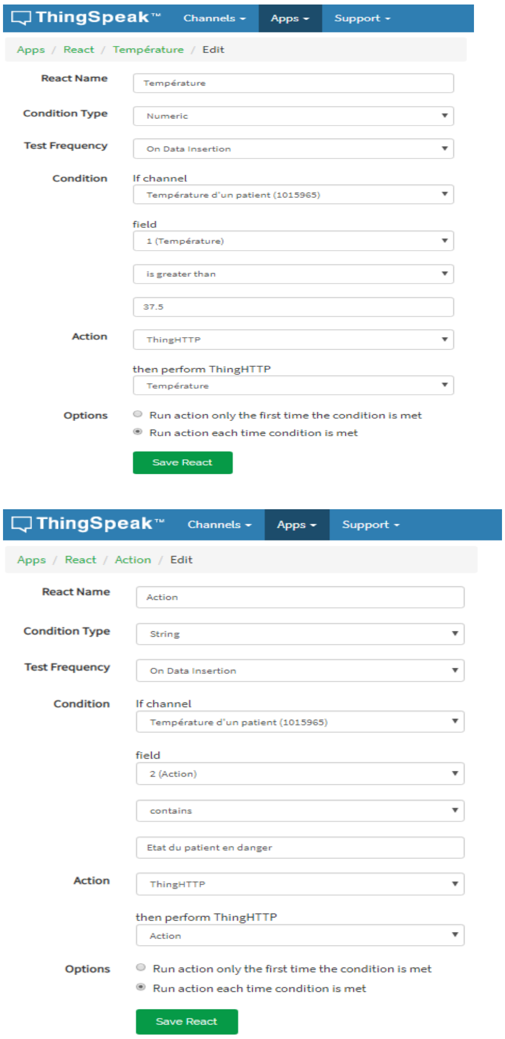


Name	Created
Température	2020-03-11
Action	2020-03-11

Fig. 8. Window showing communications between ThingHTTP and IFTTT app

2.4. IFTTT:

IFTTT is a separate business from Google. Its name means "If This Then That".Your custom command and action form a whole called an "applet".



React Name: Température
Condition Type: Numeric
Test Frequency: On Data Insertion
Condition: If channel: Température d'un patient (1015965)
field: 1 (Température)
is greater than
37.5
Action: ThingHTTP
then perform ThingHTTP: Température
Options:
○ Run action only the first time the condition is met
● Run action each time condition is met

React Name: Action
Condition Type: String
Test Frequency: On Data Insertion
Condition: If channel: Température d'un patient (1015965)
field: 2 (Action)
contains
Etat du patient en danger
Action: ThingHTTP
then perform ThingHTTP: Action
Options:
○ Run action only the first time the condition is met
● Run action each time condition is met

Fig. 9. Windows of IFTTT applets

3 PROPOSED SYSTEM

The LM35DZ temperature sensor is powered by 3.3v generated by the NodeMcu board, the analog value Vin image of the temperature is sent to the Node board for processing.

The transfer of data between the NodeMcu card and the computer must go through the ThingSpeak site which is a website for the IoT. It is an interface that is used to connect software to services, to collect, store and analyze data.

4 HARWARD REALIZATION

In The Figure 10 shows the components of an "object detection and monitoring" system that includes: The LM35DZ temperature sensor, an NodeMcu.

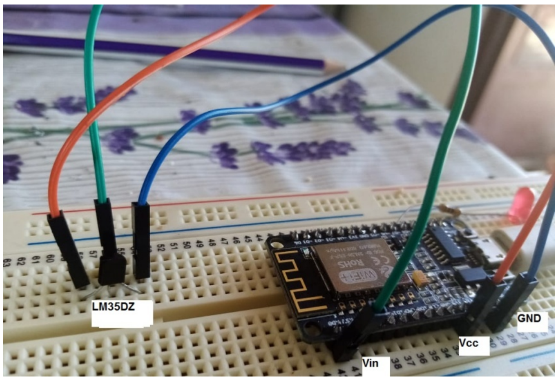
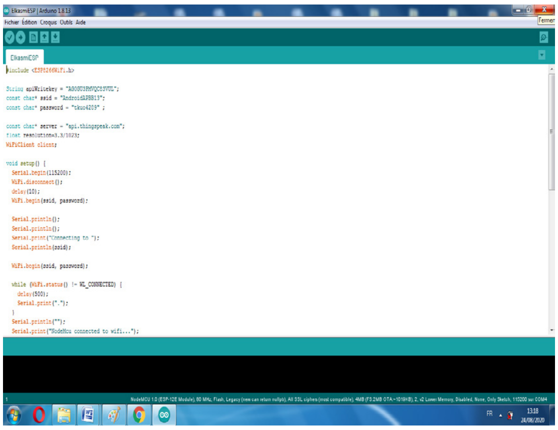


Fig. 10. Components of the remote monitoring system of an object in real time

4.1. Software Part (Code)


ElkasmiESP | Arduino 1.8.13
Fichier Édition Croquis Outils Aide
Fermer
ElkasmiESP
FR
13:18
24/08/2020

```
#include <ESP8266WiFi.h>

String apiWritekey = "AGOGU3PMVQC83VUL";
const char* ssid = "AndroidAPBB13";
const char* password = "tkuo4209" ;

const char* server = "api.thingspeak.com";
float resolution=3.3/1023;
WiFiClient client;

void setup() {
  Serial.begin(115200);
  WiFi.disconnect();
  delay(10);
  WiFi.begin(ssid, password);

  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("NodeMcu connected to wifi...");
```

```
1        NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Legacy (new can return nullptr), All SSL ciphers (most compatible), 4MB (FS:2MB OTA:~1019KB), 2, v2 Lower Memory, Disabled, None, Only Sketch, 115200 sur COM4
```

Fig. 11. main program code with arduino IDE

4.2. System Implementation

The operation of the measuring system is based on the developed program which is executed by the NodeMcu card, which will then deliver all the measurements obtained each determined time interval.

The next is to send the measurements taken via wifi to the ThingSpeak site to be saved and displayed as graphs. The flowchart for the transmission of data from the NodeMcu card to the computer is shown in the figure.

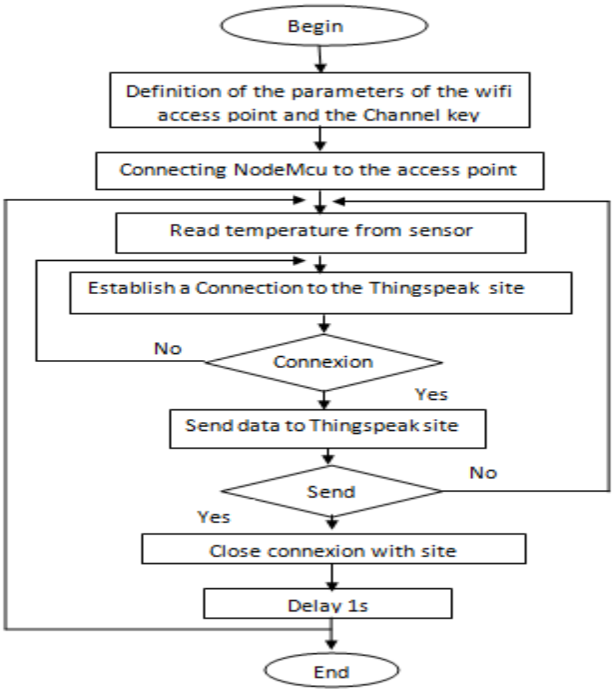
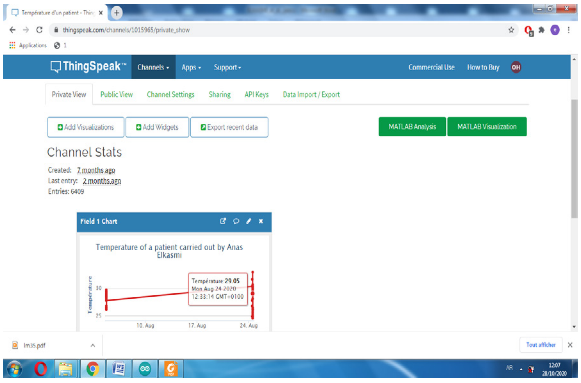

Begin
Definition of the parameters of the wifi access point and the Channel key
Connecting NodeMcu to the access point
Read temperature from sensor
Establish a Connection to the Thingspeak site
No
Connexion
Yes
Send data to Thingspeak site
No
Send
Yes
Close connexion with site
Delay 1s
End

Fig. 12. flowchart for the transmission of data.

5 RESULTS AND DISCUSSIONS

5.1. Temperature Measurement

When turn on the energy, the system consists in sending the measurements carried out via wifi to the ThingSpeak site to be saved and displayed as graphs.

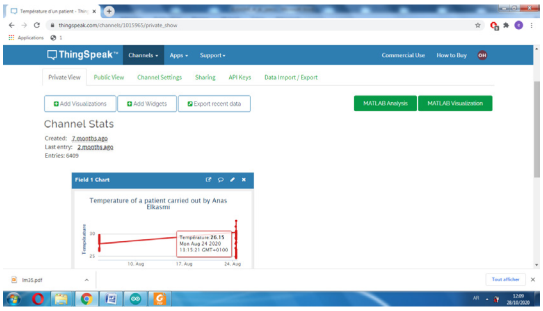


Channel Stats

Created: 7 months ago
Last entry: 2 months ago
Entries: 6409

Température: 29.05
Mon Aug 24 2020
12:33:14 GMT+0100

Fig. 13. Window of channel state on thingspeak.com site at the beginning of measurements

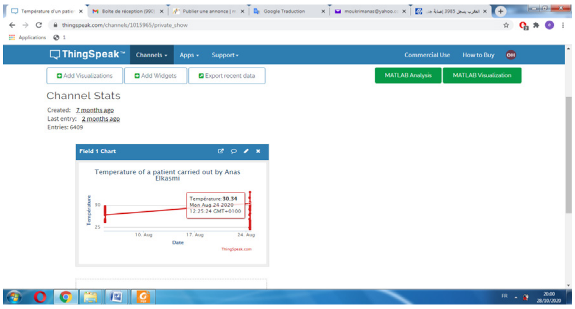


Channel Stats
Created: 7 months ago
Last entry: 2 months ago
Entries: 6409

Temperature of a patient carried out by Anas Elkasmi

Température: 26.15
Mon Aug 24 2020
13:15:21 GMT+0100

Fig. 14. Window of channel state on thingspeak.com site at other times



Channel Stats
Created: 7 months ago
Last entry: 2 months ago
Entries: 6409

Temperature of a patient carried out by Anas Elkasmi

Température: 30.34
Mon Aug 24 2020
12:25:24 GMT+0100

Fig. 15. Window of channel state on thingspeak.com site at other times

5.2. Discussions

The first series of tests with the NodeMCU assembled with Thingspeak server allowed us to obtain the following curve: the delay for taking measurements is carried out regularly by setting the "Delay" loop from the Arduino code one second for our example.

6 CONCLUSION

In this work we sought to design a system for measuring the temperature of a patient. It is composed of a temperature sensor, a NodeMcu microcontroller card whose the role is data processing and sending measurements to the ThingSpeak site for be recorded and displayed as graphs on a computer. As perspectives of this work, extensions of the measurement system can be considered taking into account other health variables such as blood pressure measurement and other measurements.

ACKNOWLEDGEMENTS

The authors are very much thankful to the unanimous reviewers of the paper and editors of the journal for their constructive and helpful comments that improved the quality of the paper.

REFERENCES

1. E. Teaw, et al., "A Wireless Health Monitoring System," International Conference on Information Acquisition, 2005, June 27- July 3.
2. P. Gupta *et al.,* "An IoT Framework for Addressing Parents Concerns about Safety of School Going Children," *International Journal of Electrical and computer Engineering (IJECE),* vol. 6, n°6, pp. 3052, 2016.
3. N. A. A. Rahman, *et al.*, "Biomedical health monitoring system design and analysis," *Indonesian Journal of Electrical Engineering and Computer Science, Vol 13,* pp. 1056-1064, 2019.
4. B. Li, *et al.*, "Research and Application on the Smart Home Based on Component Technologies and Internet of Things," Procedia Engineering, *Vol 15,* pp. 2087-2092, 2011.
5. S. Kumar, *et al.*, "Ubiquitous Smart Home System Using Android Application," *International journal of Computer Networks & Communications,* vol. 6, no 1, pp. 33-43, janv. 2014.
6. V. A. Memos, et al., " An Efficient Algorithm for Media-based Surveillance System (EAMSuS) in IoT Smart City Framework," *Future Generation Computer Systems,* vol. 83, pp. 619-628, 2018.
7. P. S. Saarika, *et al.*, " Smart transportation system using IoT," *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), Bangalore,* pp. 1104-1107, 2017.
8. V. Suryani,, *et al.*, " ConTrust: A Trust Model to Enhance the Privacy in Internet of Things," International Journal of Intelligent Engineering and Systems, *Vol 10 n°3*, pp. 30-37, 2017.
9. R. Habachi, *et al.*, " Resolution of economic dispatch problem of the morocco network using crow search algorithm," *Indonesian Journal of Electrical Engineering and Computer Science,* vol. 13, n°1, pp. 347, janv. 2019.
10. K. Radharamana, *et al.*, " Web Architecture for Monitoring Field using Representational State Transfer Methods," *International Journal of Intelligent Engineering and Systems,* vol. 12, pp. 84-93, 2019.
11. P. Asghari,, *et al.*, " Internet of Things applications: A systematic review," *Computer Networks, Vols 148,* pp. 241-261, janv 2019

6

BIOGRAPHIES OF AUTHORS

	Abdelkader Boulezhar Professor, Faculty of sciences Ain Chock, Physics, Hassan II University- Casablanca Morocco.
	Mostapha Abouricha Professor, Faculty of Sciences Ibn Zohr, Physics, University of Agadir
	Anas El Kasmi Phd Student, Faculty of sciences Ain Chock, Physics, Hassan II University- Casablanca Morocco.