Final Project

Development and Editing: Zacharenia Garofalaki

Target

Confirmation of knowledge on topics: microcontrollers and study of hardware resources, initial installation, serial communication activities, digital and analog inputs/outputs, use and wiring of electronic components, IoT platforms and hardware networking, data exchange between hardware and platform, study of communication protocols and available APIs, graphical data visualization.

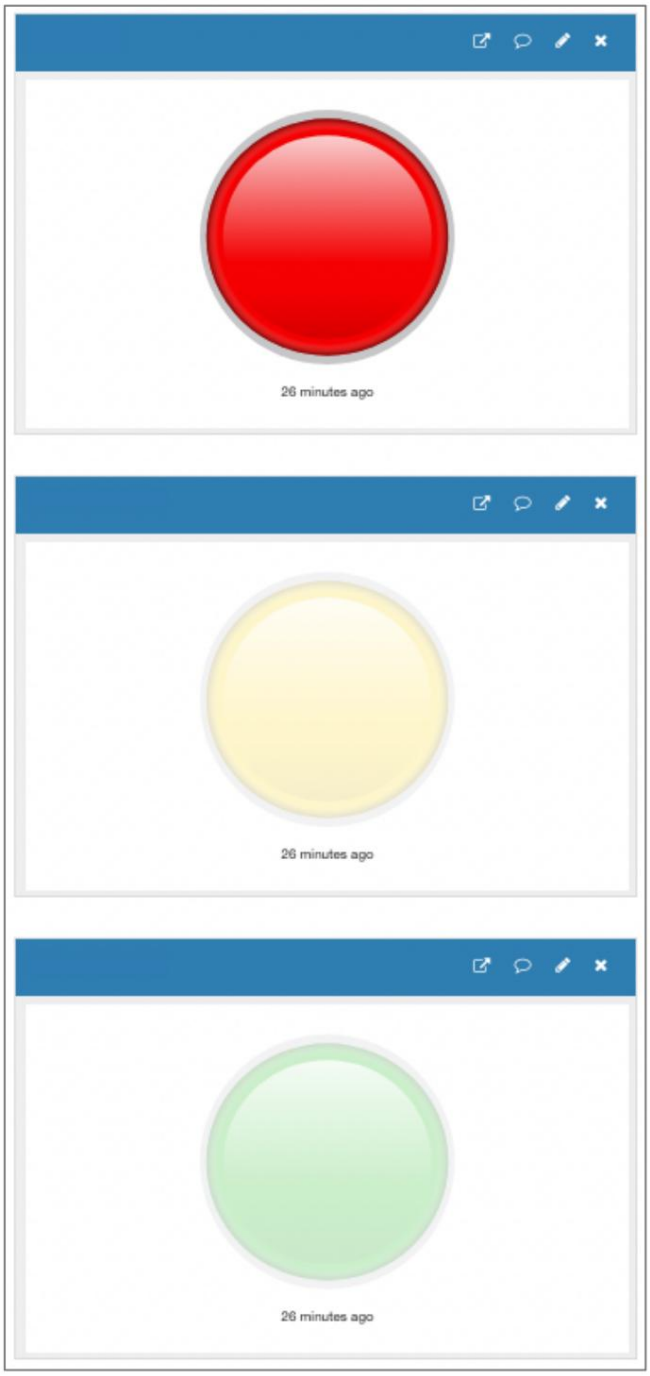

Figure 1 Visual indications of traffic light operation on the IoT platform

Wanted

A. Electric traffic light

You are invited to create a traffic light whose function will be defined by the Arduino microcontroller and will be displayed on the IoT platform ThingSpeak. The traffic light should operate continuously, with the alternating signals following the sequence:

(a) Red indication, duration >=30secs

(b) Green indicator, duration >=30secs

(c) Orange indication, duration >=20secs

(d) Red indication, duration >=30secs

(e) ….etc.

For this activity you will need to implement the following:

1. Use of visual elements in the ThingSpeak channel to illustrate the above desired functionality. Figure 1 shows an illustrative case of using visual elements.

2. Connecting the above visual elements to a variable or variables of the channel.

3. Connecting the Arduino to the nearest WiFi AP using the ESP-01.

4. Programming the Arduino to operate the visual elements in ThingSpeak, in accordance with the above requirements of the Activity.

B. Sending data to another application's channel

You are invited to integrate into the code of Request A.4, the **ability to set a value for a variable,** let it be the Field 8 variable, of the **channel of another application** (i.e. the channel of another traffic light, of another work group). For this purpose, you will need the channel details of the other application, which you should report in your implementation and deliverables. For better control of your application, the changes that will be applied to Field8 of the other application's channel, should **also be applied to Field8 of your own channel.**

For this specific activity you will need to implement:

1. Programming code that implements what the code of Request A.4 implemented and **in addition** to setting the value 0 to the **Field 8 channel variable of another application.**

2. Programming code that implements what the code in Request A.4 implemented and **also** sets the value 0 to the **Field 8 variable of your channel.**

NOTE: Field 8 refers here to one of the 8 variables provided in a ThingSpeak channel. For more information, refer to Lab 2.

C. Reading data on another application's channel

You are invited to incorporate into the code of Request B.2, the **ability to read the value of a variable,** let's say the Field 8 variable. This specific variable is already defined by the code of Request B.2 to have a value of 0. Here Field 8 is used as a notification variable, the

which will always have a value of 0. Although as long as it has a value of 1, the traffic light should be temporarily out of operation, that is, it should have an orange light indicator.

For this specific activity you will need to implement:

1. Programming code that implements what the code of Request B.2 implemented and **in addition** sets an **orange** light indicator on the traffic light when **Field 8 acquires a value of 1** and until it returns to a value of 0.
2. Add a setting to the programming code of Request C.1, so that Field 8 acquires value 1 **every 10 minutes** and this lasts **for 1 minute.**

Deliverables

For the above activities you will need to submit:

1. **Text file** with: description of your application and its functions, illustration and explanation of the hardware circuit, description of the ThingSpeak channel settings (visual elements used, variable settings, etc.), analysis of the code you have written. The text file should contain a clear reference to the individual Activities.

2. ***.ino files** with the programming code of Requests A.4, B.1, B.2, C.1 and C.2.

The above files should be submitted in a compressed file with an indicative file name **AAAAA_BBBBBB.zip,** where AAAAA and BBBBB are the AMs of the working group members.

REMINDER that the results of the Final Project will be presented by the members of the working group in the laboratory. The submission of the deliverables and the presentation are required for the evaluation of the members of the working group. The above also applies to those belonging to the **[overflow] section.**